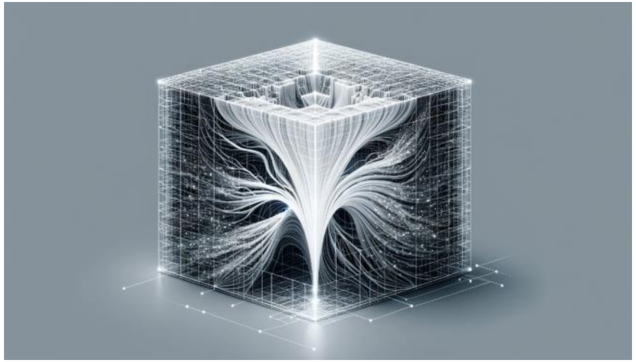
DEMOTE: Dynamic Tensor Decomposition via Neural Diffusion-Reaction Processes

NeurIPS 2023 (Spotlight)

Zheng Wang*, Shikai Fang*,
Shibo Li, Shandian Zhe
Presenter: Shikai Fang
April 2024 @ TensorNet Reading Group, Mila

KAHLERT SCHOOL OF COMPUTING

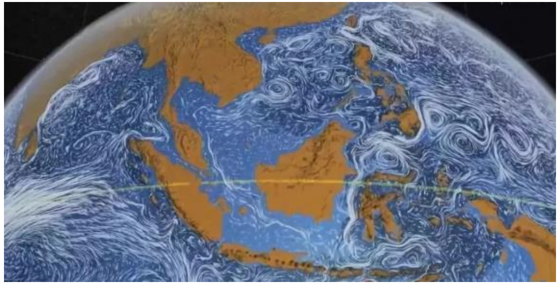


Tensor Data

- Multi-dim array for **high-order structural** data

Entry: (index1, index2..)-> value ⇔ Interaction of multiple objects

Climate System



(region, topography, weather)

Online Ads

(user, movie, site, device)

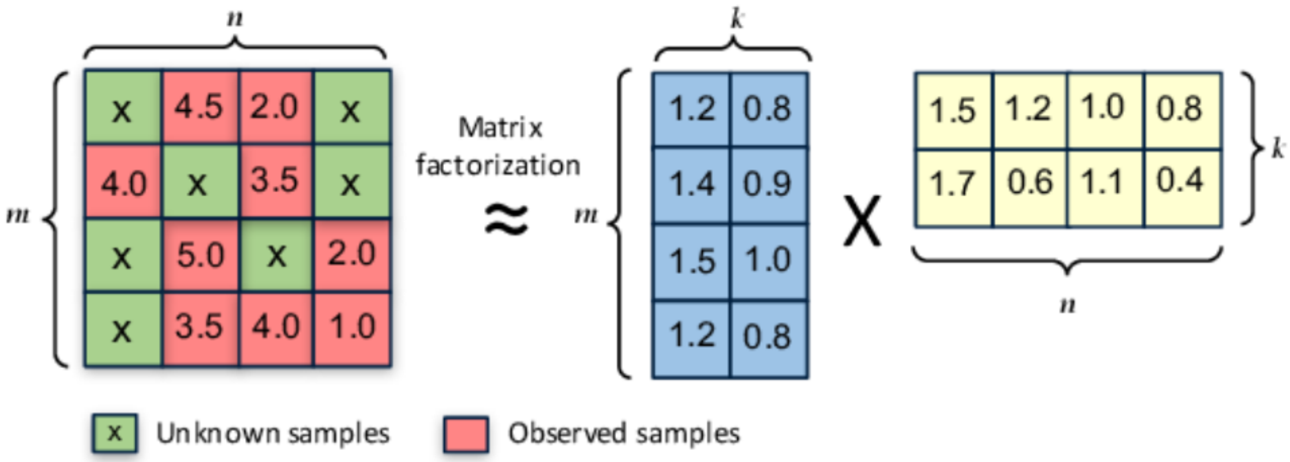
Traffic Flow

(city, road, population, period)

Tensor Decomposition

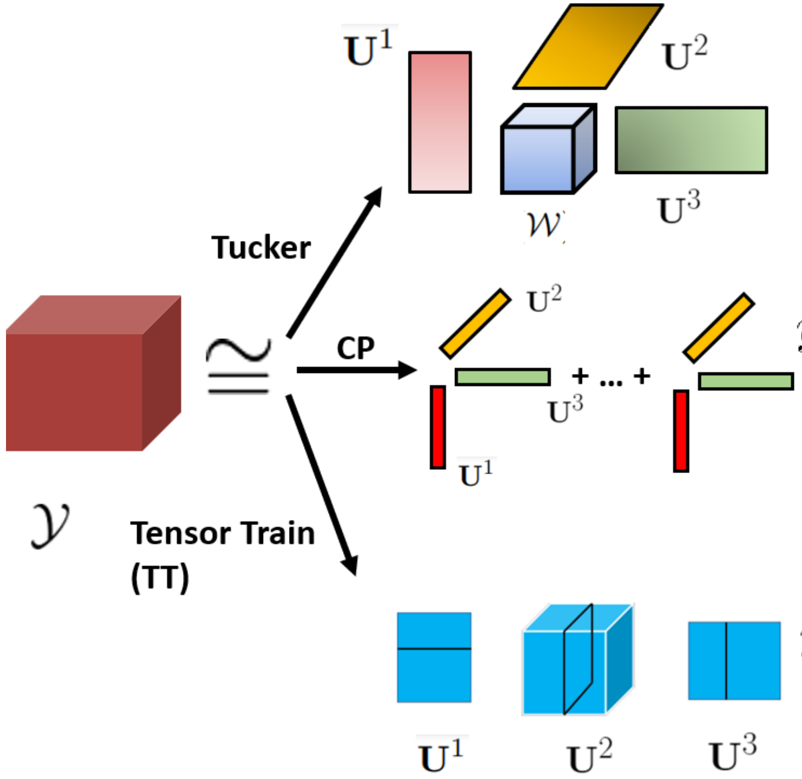
- Learn **low-rank** factors(embeddings) of high-order tensor
- 2-D case: **Collaborative Filterling** (Matrix Factorization)

Rating Matrix **User Factors** **Moive Factors**



NETFLIX

CP / Tucker / TT Decomposition



$$y_{\mathbf{i}} \approx \mathrm{vec}(\mathcal{W})^\top \left(\mathbf{u}^1_{i_1} \otimes \ldots \otimes \mathbf{u}^K_{i_K}\right)$$

$$y_{\mathbf{i}} \approx \left(\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K}\right)^\top \mathbf{1}$$

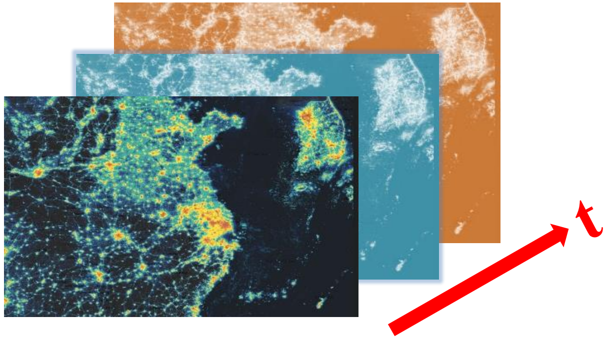
$$y_{\mathbf{i}} \approx \underbrace{\mathbf{u}^1_{i_1}}_{r_0 \times r_1} \underbrace{\mathbf{u}^2_{i_2}}_{r_1 \times r_2} \cdots \underbrace{\mathbf{u}^K_{i_K}}_{r_{K-1} \times r_K}$$

Low-rank factors

$$\mathbf{U}^k = [\mathbf{u}^k_1 \ldots \mathbf{u}^k_{d_k}]$$

Temporal Tensor Data

- **Tensor-valued time series**



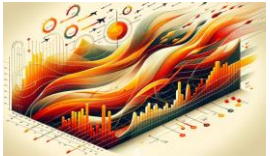
(region, site, weather) × time

(user, user,location, message) × time

Tensor structure are evolving through time!

Example: Wind Power Prediction

- **Tensor-valued time series:**

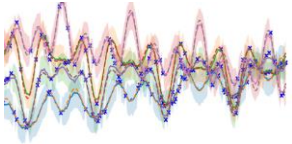


Climate mode

Turbine Life

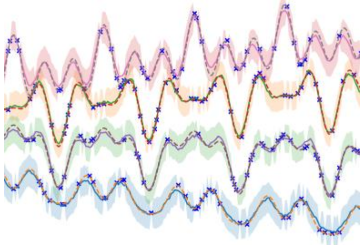
Ocean current

Seq. to Seq.
(classical time series)



Historic Seq.

?

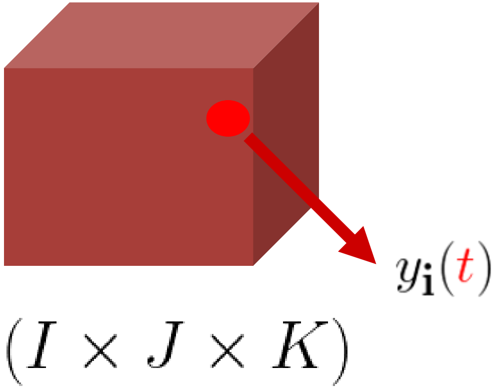


Temporal Tensor!

(climate, turbine, current) $\times$ time –> **power(**t**)**

Temporal Tensor: trival solutions

- Drop timestamps / Aggregate over time?

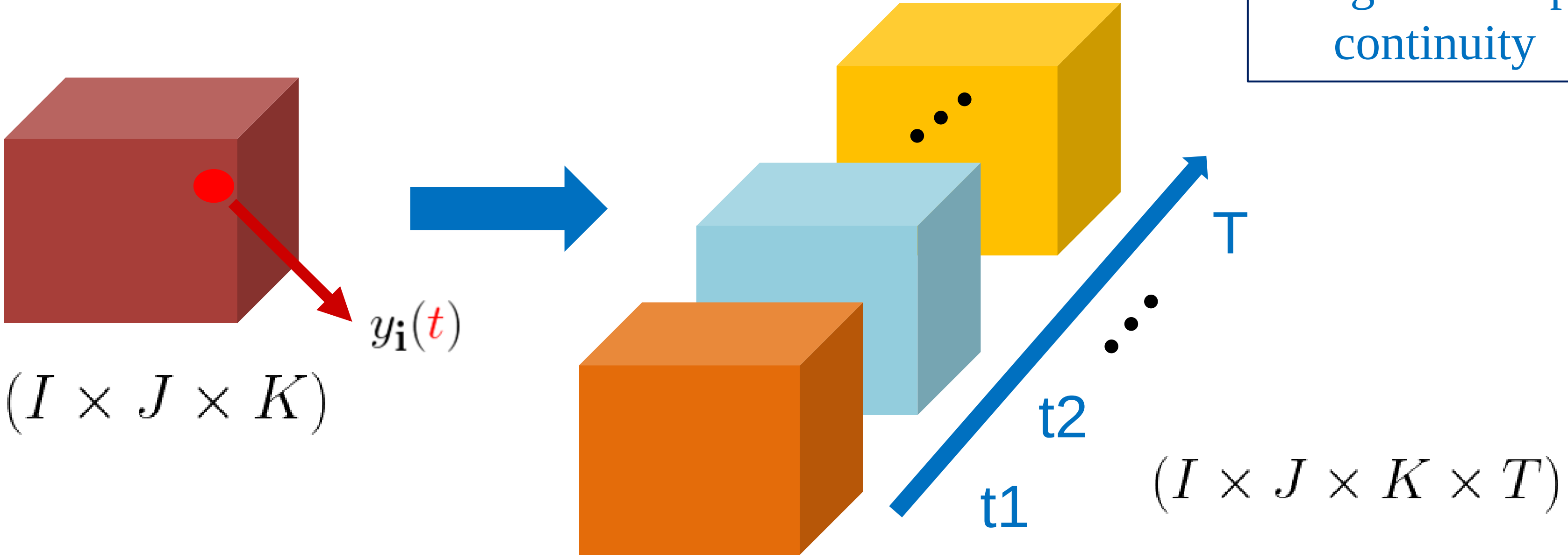


--losing temporal info

Temporal Tensor: trival solutions

- Augment tensor with discrete time mode

- Too Sparse!
- Ignore temporal continuity



Current Temporal Tensor model

Most existing work[1][2][3]:

Evolving interaction weights + Static factors

$$y_{\mathbf{i}}(t) \approx \mathbf{w}(t)^\top \left(\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K}\right)$$

- Easy to inference, but **over-simplified**!
- **Evolving factors** dominate in many cases

[1] Zhang, Yanqing, et al. "Dynamic tensor recommender systems." *The Journal of Machine Learning Research* 22.1 (2021): 3032-3066.
[2] Fang, Shikai, et al. "Bayesian Continuous-Time Tucker Decomposition." *International Conference on Machine Learning*. PMLR, 2022.
[3] Li, Shibo, et al.. "Decomposing Temporal High-Order Interactions via Latent ODEs." *International Conference on Machine Learning*. PMLR, 2022.

Motivation of our work

Goal: Learning dynamic factor trajectories!

$$y_{\mathbf{i}}(t) \approx g\left(\mathbf{u}^1_{i_1}(t), \mathbf{u}^2_{i_2}(t) \ldots, \mathbf{u}^K_{i_K}(t)\right)$$

Time-varing represent. of real-world object!

Motivation of our work

Goal: Learning dynamic factor trajectories!

$$y_{\mathbf{i}}(t) \approx g\left(\mathbf{u}^1_{i_1}(t), \mathbf{u}^2_{i_2}(t) \ldots, \mathbf{u}^K_{i_K}(t)\right)$$

Challenges:

- **How to model the dynamic factors in an efficient and flexiable way?**

- **How to model the possible dependency over co-evolving factors?**

Insight: Bridge of Tensor and Graph

$$y_{\mathbf{i}}(t) \approx g\left(\mathbf{u}^1_{i_1}(t), \mathbf{u}^2_{i_2}(t) \ldots, \mathbf{u}^K_{i_K}(t)\right)$$

- How to model the possible dependency over co-evolving factors?

Build a Graph!

High-level insights:

- We know **graph <=> matrix**...
- **Tensor data** <=> **Hyper-graph / K-Partite graph (encoding inherent nature of tensor)**
- Dynamic factors <=> graph-constrained ODE / dynamic process on graph!

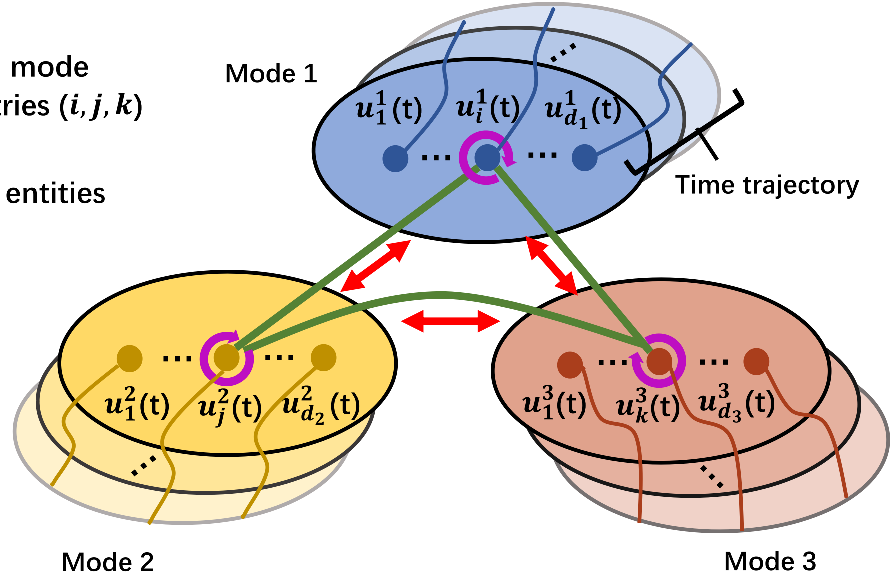
Our Solution: DEMOTE

Key Idea: dynamic tensor <=> dynamic process on a (K-Partite)-graph

- : Embeddings of entities of each mode
- : Edges defined by observed entries (i, j, k)
- : Diffusion process along edges
- : Reaction process on individual entities

Table of observed data

Mode 1	Mode 2	Mode 3	Time-stamp	Value
i	j	k	t	$y_{ijk}(t)$
…	…	…	…	…



Our Solution: DEMOTE

How to build a graph from tensor data?

- Graph node ⇔ tensor factors
- Graph edges ⇔ tensor entries (interaction of factors)
- Edge weights W: learnable

: Embeddings of entities of each mode

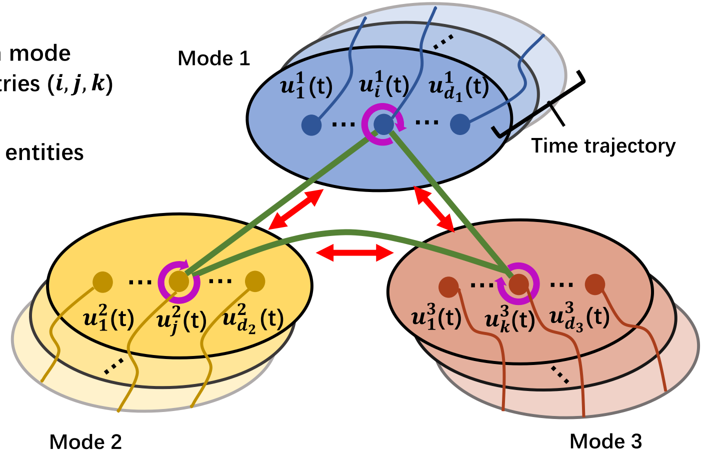
: Edges defined by observed entries (i, j, k)

: Diffusion process along edges

: Reaction process on individual entities

Table of observed data

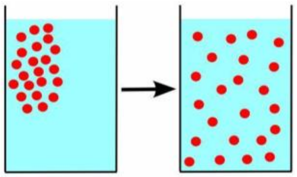
Mode 1	Mode 2	Mode 3	Time-stamp	Value
i	j	k	t	$y_{ijk}(t)$
…	…	…	…	…



Our Solution: DEMOTE

How to model the dynamic process?

Diffusion process on graph



Model concentration change on the graph! and co-evolving of factors

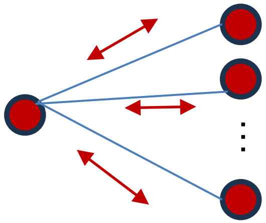
Node-wise:

$$\frac{\mathrm{d}\mathbf{u}_j^k}{\mathrm{d}t} = \sum_{s\in\{1,\dots,K\}\setminus k}\sum_{j'=1}^{d_s}[\mathbf{W}^{k,s}]_{j,j'}\left(\mathbf{u}^s_{j'}(t)-\mathbf{u}^k_j(t)\right) = \sum_{s\in\{1,\dots,K\}\setminus k}\left(\mathbf{w}_j^{k,s}\mathbf{U}^s(t)\right)^\top - a_j^{k,s}\mathbf{u}_j^k,$$

Diffusion ratio

Edge weights (strength of connection)

“concertration gap”



Group-wise:

$$\frac{\partial\mathcal{U}(t)}{\partial t} = (\mathcal{W}-\mathcal{A})\mathcal{U}(t)$$

$$\mathcal{W} = \begin{pmatrix} \mathbf{0} & \mathbf{W}^{1,2} & \dots & \mathbf{W}^{1,K} \\ \mathbf{W}^{2,1} & \mathbf{0} & \dots & \vdots \\ \vdots & & \ddots & \mathbf{W}^{K-1,K} \\ \mathbf{W}^{K,1} & \dots & \mathbf{W}^{K,K-1} & \mathbf{0} \end{pmatrix}$$

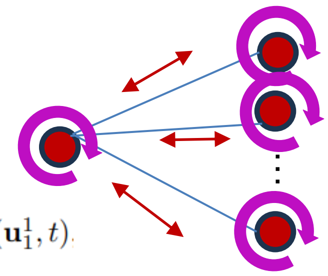
$$\mathcal{A} = \mathrm{diag}\left(\sum\nolimits_{s\in\{1\dots K\}\setminus 1}\mathbf{A}^{1,s},\dots,\sum\nolimits_{s\in\{1\dots K\}\setminus K}\mathbf{A}^{K,s}\right)$$

Our Solution: DEMOTE

How to model the dynamic process?

Diffusion process on graph + **Reaction Process**

Model the local concentration change and self-envoving of factors



$\mathbf{f}_{\boldsymbol{\theta}_1}(\mathbf{u}_1^1, t)$

Final diffusion-reaction ODE:

$$\frac{\partial \mathcal{U}(t)}{\partial t} = (\mathcal{W} - \mathcal{A})\mathcal{U}(t) + \mathcal{F}(\mathcal{U}, t), \quad \mathcal{U}(0) = \mathcal{U}_0$$

$$\mathcal{F}(\mathcal{U}, t) = [\mathbf{f}_{\boldsymbol{\theta}_1}(\mathbf{u}_1^1, t), \ldots, \mathbf{f}_{\boldsymbol{\theta}_1}(\mathbf{u}_{d_1}^1, t), \ldots, \mathbf{f}_{\boldsymbol{\theta}_K}(\mathbf{u}_1^K, t), \ldots, \mathbf{f}_{\boldsymbol{\theta}_K}(\mathbf{u}_{d_K}^K, t)]^\top.$$

Our Solution: DEMOTE

Entrt value Generation: nonlinear NN-based decomposition

$$y_\ell(t) = g\left(\mathbf{u}^1_{l_1}(t), \ldots, \mathbf{u}^K_{l_K}(t)\right)$$

Joint Probability:

$$p(\boldsymbol{\beta}, \{\boldsymbol{\theta}_k\}, \mathbf{y}) = \underbrace{p(\boldsymbol{\beta})}_{\text{Gaussian Priors}} \cdot \prod_{k=1}^{K} \underbrace{p(\boldsymbol{\theta}_k)}_{\text{Gaussian Prior}} \cdot \prod_{n=1}^{N} \underbrace{\mathcal{N}\left(y_n | g\left(\mathbf{u}^1_{l_{n1}}(t_n), \ldots, \mathbf{u}^K_{l_{nK}}(t_n)\right), \sigma^2\mathbf{I}\right)}_{\text{Gaussian Likelihood}}$$

Para. of g ($\boldsymbol{\beta}$)

Paras. of reaction ($\{\boldsymbol{\theta}_k\}$)

Gaussian Priors

Gaussian Prior

Gaussian Likelihood

Inference of DEMOTE

- Run **gradient-track ODEsolver** first:

$$\frac{\partial \mathcal{U}(t)}{\partial t} = (\mathcal{W} - \mathcal{A})\mathcal{U}(t) + \mathcal{F}(\mathcal{U}, t), \quad \mathcal{U}(0) = \mathcal{U}_0 \longrightarrow \mathcal{U}(t_n) = \text{ODESolve}(\mathcal{U}_0, 0, t_n, \Theta)$$

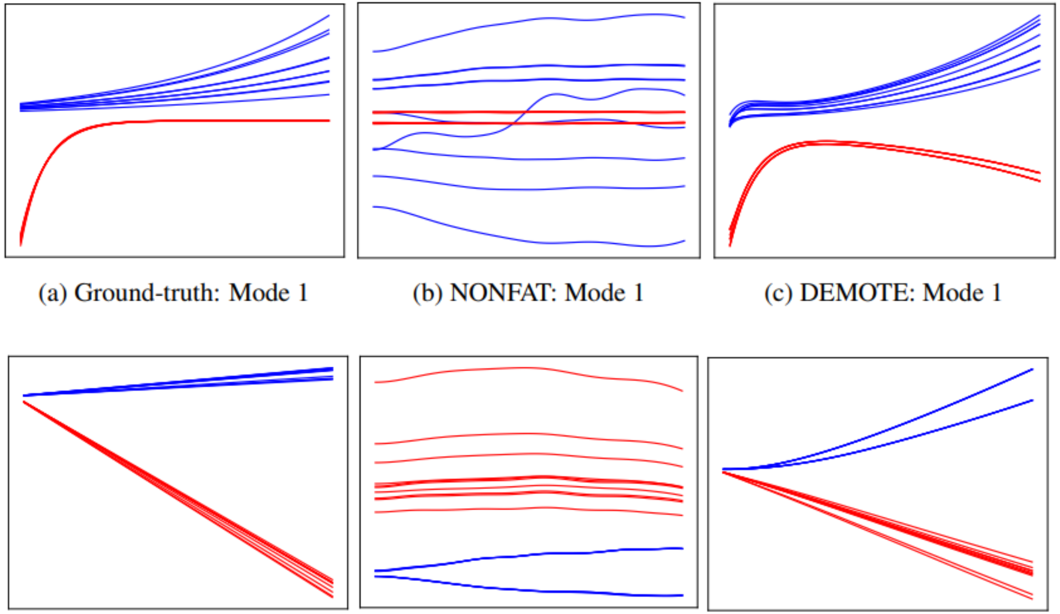
Gradient of edge weights,
initial value,
parameters of reaction process

- Maximize the log-joint probability by **mini-batch stochastic optimization**

$$\mathcal{L} = \log p(\boldsymbol{\beta}, \{\boldsymbol{\theta}_k\}, \mathbf{y}) = \log(\text{Prior}) - \sum_{n=1}^{N} \log \mathcal{N}\left(y_n | g\left(\mathbf{u}^1_{l_{n1}}(t_n), \ldots, \mathbf{u}^K_{l_{nK}}(t_n)\right), \sigma^2 \mathbf{I}\right)$$

$$\log(\text{Prior}) = \log p(\boldsymbol{\beta}) + \sum_{k=1}^{K} \log p(\boldsymbol{\theta}_k).$$

Evaluation of DEMOTE: synthetic task



(a) Ground-truth: Mode 1 (b) NONFAT: Mode 1 (c) DEMOTE: Mode 1

(d) Ground-truth: Mode 2 (e) NONFAT: Mode 2 (f) DEMOTE: Mode 2

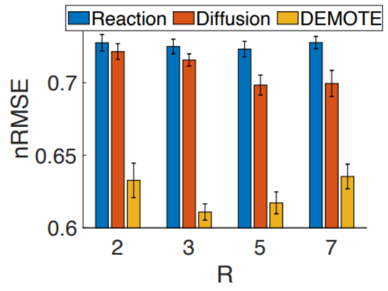
$$u_j^1(t) = c_j^1 \exp\left(0.5 c_j^1 t\right)$$
$$(1 \le j \le 20)$$

$$u_j^2(t) = c_j^2 + 2\pi c_j^2 t$$
$$(1 \le j \le 20)$$

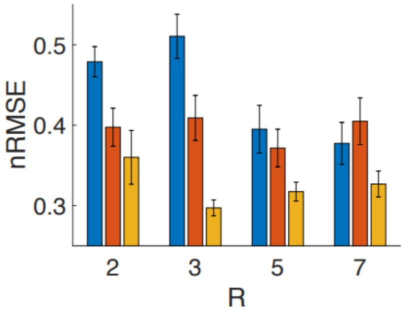
Predictive performance & ablation study

CA Weather	$R=2$	$R=3$	$R=5$	$R=7$
CP-DTLD	0.7440 ± 0.0035	0.7372 ± 0.0040	0.7290 ± 0.0042	0.7270 ± 0.0044
GP-DTLD	0.7417 ± 0.0031	0.7414 ± 0.0036	0.7444 ± 0.0036	0.7449 ± 0.0039
NN-DTLD	0.7228 ± 0.0054	0.7116 ± 0.0033	0.7070 ± 0.0041	0.7065 ± 0.0038
CP-DTND	0.7448 ± 0.0031	0.7360 ± 0.0035	0.7273 ± 0.0037	0.7280 ± 0.0044
GP-DTND	0.7399 ± 0.0034	0.7346 ± 0.0032	0.7448 ± 0.0037	0.7467 ± 0.0031
NN-DTND	0.7113 ± 0.0045	0.6979 ± 0.0126	0.6659 ± 0.0122	0.6543 ± 0.0155
CP-CT	1.0000 ± 0.0096	0.9959 ± 0.0067	1.0010 ± 0.0017	1.0060 ± 0.0034
GP-CT	0.7433 ± 0.0038	0.7354 ± 0.0027	0.7359 ± 0.0034	0.7377 ± 0.0033
NN-CT	0.8697 ± 0.0014	0.8679 ± 0.0022	0.8676 ± 0.0018	0.8695 ± 0.0016
NONFAT	0.7444 ± 0.0042	0.7460 ± 0.0032	0.7645 ± 0.0061	0.7553 ± 0.0029
THIS-ODE	0.7511 ± 0.0052	0.7539 ± 0.0041	0.7614 ± 0.0024	0.7620 ± 0.0032
DEMOTE	$\mathbf{0.6327 \pm 0.0119}$	$\mathbf{0.6109 \pm 0.0056}$	$\mathbf{0.6172 \pm 0.0075}$	$\mathbf{0.6354 \pm 0.0085}$

CA Traffic				
CP-DTLD	0.6498 ± 0.0257	0.6424 ± 0.0266	0.6436 ± 0.0268	0.6405 ± 0.0262
GP-DTLD	0.6309 ± 0.0167	0.6290 ± 0.0185	0.6383 ± 0.0204	0.6496 ± 0.0193
NN-DTLD	0.6528 ± 0.0230	0.6545 ± 0.0244	0.6401 ± 0.0282	0.6136 ± 0.0338
CP-DTND	0.6497 ± 0.0245	0.6456 ± 0.0265	0.6431 ± 0.0263	0.6419 ± 0.0259
GP-DTND	0.6544 ± 0.0213	0.6559 ± 0.0224	0.6604 ± 0.0243	0.6674 ± 0.0214
NN-DTND	0.6578 ± 0.0248	0.6528 ± 0.0256	0.6519 ± 0.0249	0.6482 ± 0.0261
CP-CT	0.9858 ± 0.0120	0.9972 ± 0.0056	0.9816 ± 0.0136	0.9991 ± 0.0120
GP-CT	0.6610 ± 0.0207	0.6668 ± 0.0191	0.6756 ± 0.0190	0.6768 ± 0.0196
NN-CT	0.9804 ± 0.0017	0.9815 ± 0.0015	0.9791 ± 0.0012	0.9802 ± 0.0017
NONFAT	0.4461 ± 0.0247	0.4610 ± 0.0231	0.5031 ± 0.0155	0.6307 ± 0.0847
THIS-ODE	0.6603 ± 0.0230	0.6536 ± 0.0212	0.6838 ± 0.0193	0.6378 ± 0.0142
DEMOTE	$\mathbf{0.3601 \pm 0.0334}$	$\mathbf{0.2972 \pm 0.0099}$	$\mathbf{0.3174 \pm 0.0118}$	$\mathbf{0.3269 \pm 0.0162}$



(a) *CA Weather*



(b) *CA Traffic*

Trajactory learning for factors and entries

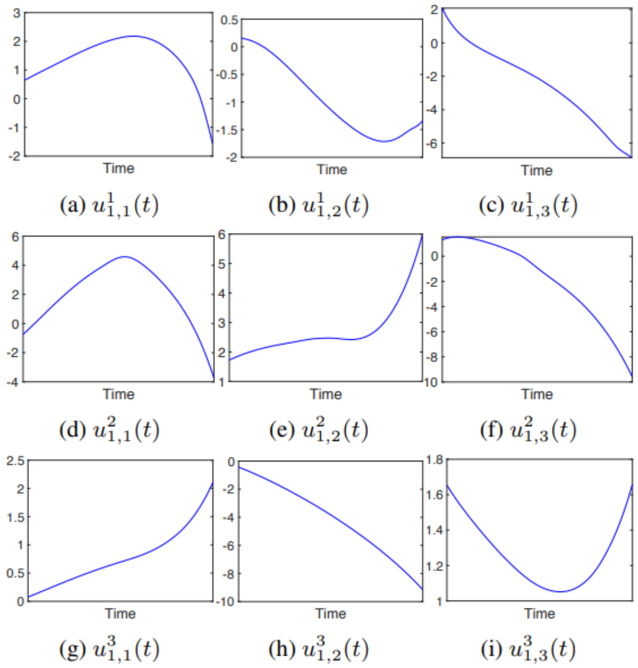


Figure 4: The learned embedding trajectories for location 1 (a-c), air conditional mode 1 (d-f), and power usage level 1 (g-i) in *Server Room* dataset.

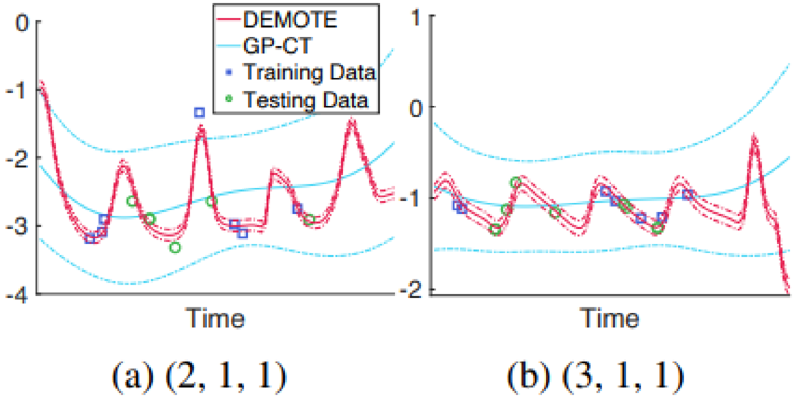
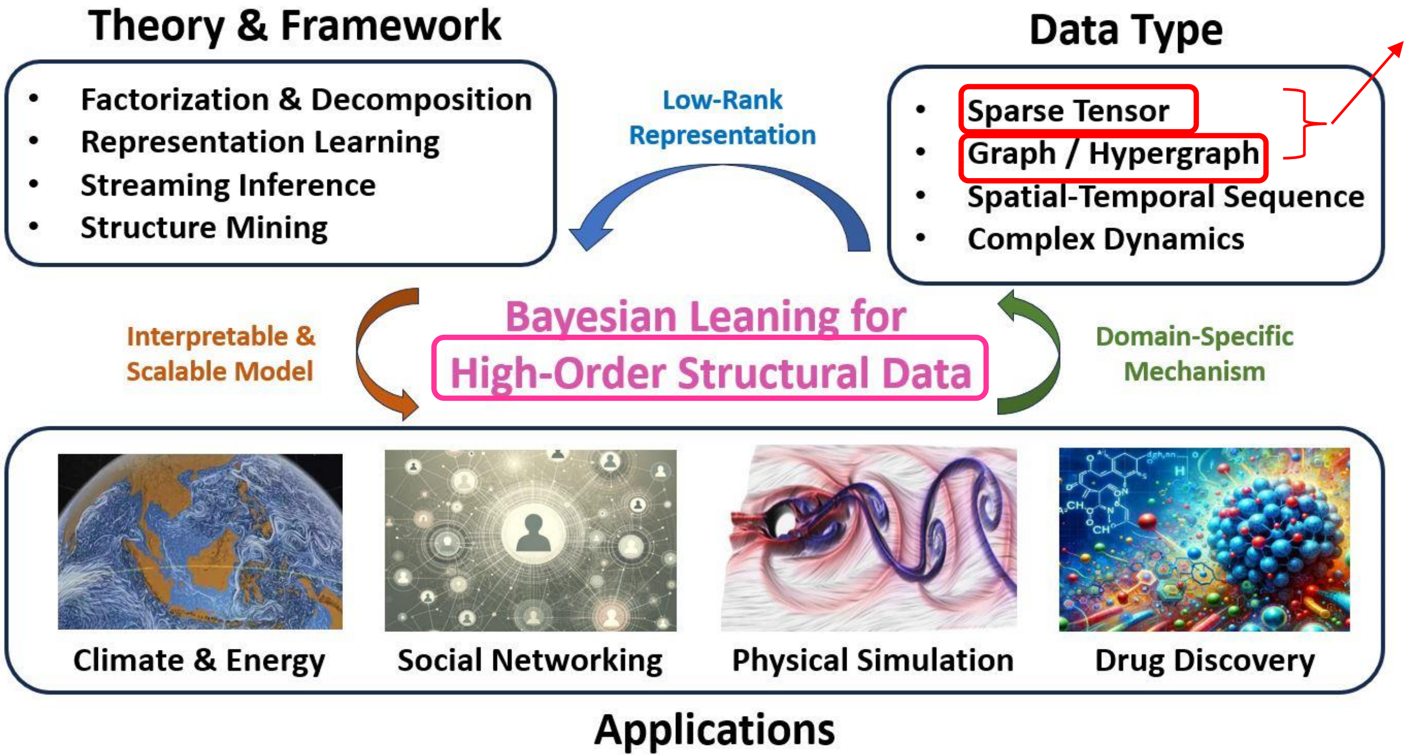


Figure 5: Entry value prediction on *Server Room*.

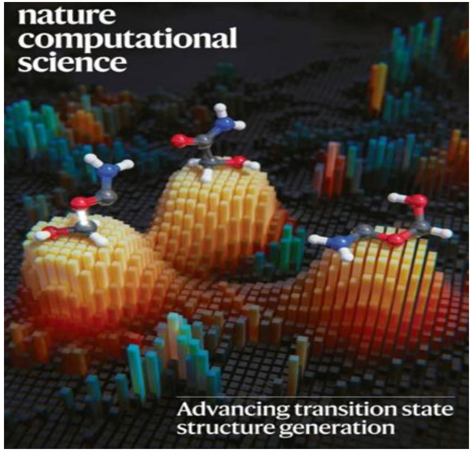
Broader view of tensor learning



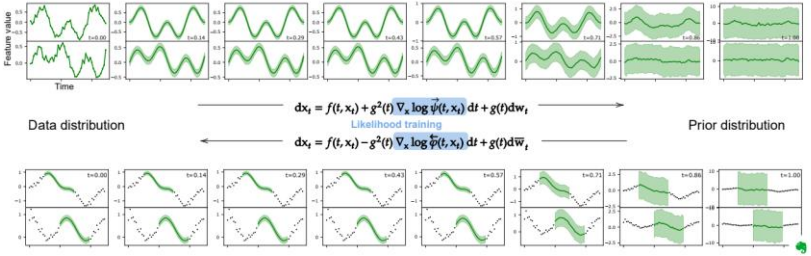
DEMOTE start here, but not the end!

Unified framework for dynamic structural Data

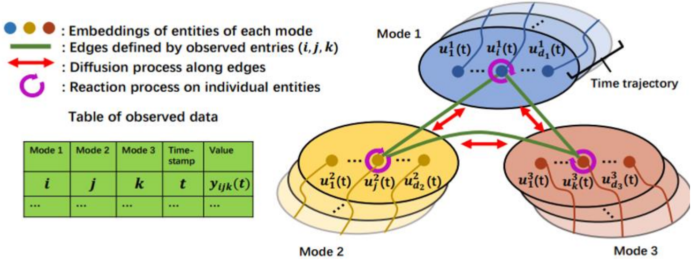
- **Low-rank** surrogate for **high-order** structure
- **Generative & Structure** in **latent** space
- **Unified framework** for various data types (tensor, hypergraph, dynamics…)



[1]



[2]



Mode 1	Mode 2	Mode 3	Time-stamp	Value
i	j	k	t	$y_{ijk}(t)$
…	…	…	…	…

[3]

[1] Duan et al., "Accurate transition state generation with an object-aware equivariant elementary reaction diffusion model", Nature Compuatational Science
[2] Chen* & **Fang*** et al., "Provably Convergent Schrodinger Bridge with Applications to Probabilistic Time Series Imputation", ICML 2023
[3] Wang* & **Fang*** et al,, "Dynamic Tensor Decomposition via Neural Diffusion-Reaction Processes", Neurips 2023

First Principles ⇔ Priors of tensor learning

- Encoding the **first principle** of science/industry as model priors

First Principle	Domain	Model with Prior
Symmetry structure	Bio, Chemistry	Equivariant GNN[1]
Diffusion process	Physics	Diffusion model[2]
Differential equation	Physics, Engineering	NeuralODE[3]
…	…	…

[4]

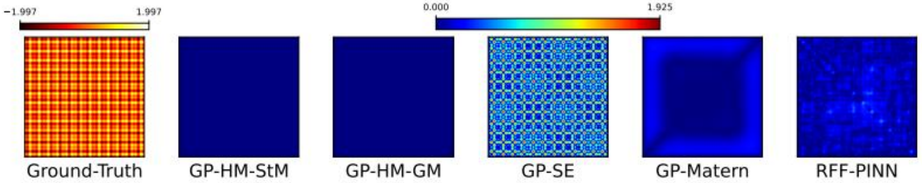


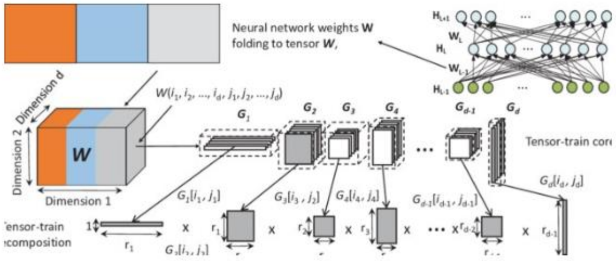
Figure 7: Point-wise solution error for 2D Poisson equation and the solution is $u(x) = \sin(6x)\sin(20x) + \sin(6y)\sin(20y)$.

[5]

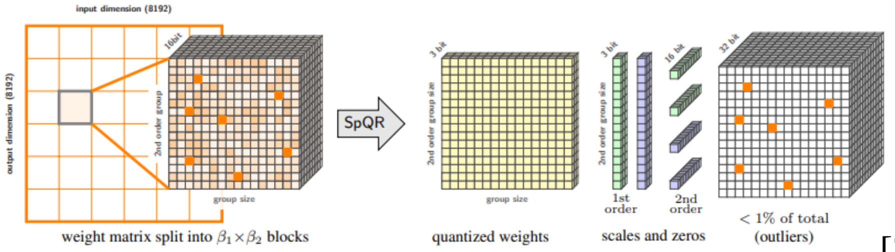
[1]:Satorras et al., “Equivariant graph neural network”, ICML 2021
[2]: Jascha, et al., "Deep unsupervised learning using nonequilibrium thermodynamics, ICML 2015
[3]: Chen et al., “Neural Ordinary Differential Equations”, NIPS 2018
[4]: Wang et al., “scientific discovery in the age of artificial”, Nature 2023
[5]: **Fang** et al., “Solving High Frequency and Multi-Scale PDEs with Gaussian Processes”, ICLR 2024

Efficient & Adaptive LLM via Low-Rank

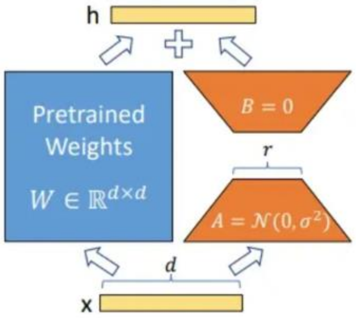
- Tensorization of **LLM + Quantization** for Edge computing
- Tensor-based **multi-LORA** for **LLM + multi-task/domain fine-tuning/alignment**



[1]



[2]



[3]

[1]: Huang et al., "An energy-efficient machine learning accelerator on 3D CMOS-RRAM f
or layer-wise tensorized neural network", *IEEE SOCC*., 2017.
[2]: Tim Dettmers et al., "SpQR: A Sparse-Quantized Representation for Near-Lossless LLM Weight Compression", arxiv 2306.03078
[3]: Edward J. Hu, et al., "LoRA: Low-Rank Adaptation of Large Language Models", arxiv 2106.0968

Thank you!

Q&A

Appendix