LSPTD: Low-rank and spatiotemporal priors enhanced Tucker decomposition for internet traffic data imputation

Prof Lili YANG

Southern University of Science and Technology, China

Citation: Wenwu Gong, Zhejun Huang, Lili Yang, “LSPTD: Low-rank and spatiotemporal priors enhanced Tucker decomposition for internet traffic data imputation”, in 2023 IEEE Conference on Intelligent Transportation Systems (ITSC 2023).

GitHub repositories: https://github.com/GongWenwuu/LSPTD.git

Background

- Internet traffic data (ITD) records subjects' movement from an origin to a destination, involving origin-destination (OD) pair, time, and date modes, which contains useful information for intelligent transportation systems.
- Unfortunately, the missing data problem frequently occurs due to communication malfunctions, transmission distortions, or adverse weather conditions [1].
- Internet traffic data imputation (ITDI) refers to estimating the missing values from observations.
- ITD accurately store spatial and temporal information and is often organized by matrices or tensors.

Roughan et al. [2] used the low rank matrix factorization to impute the missing traffic data

Tan et al. [3] used the low rank tensor completion to capture the spatial-temporal information and achieved better results

⟹ identifying the tensor structure from the raw data is a more appropriate method for ITDI

[1] X. Chen, Z. He, and L. Sun, "A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation," Transportation Research Part C: Emerging Technologies, vol. 98, pp. 73–8.
[2] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices (extended version)," IEEE/ACM Transactions on Networking, vol. 20, no. 3, pp. 662–676, 2012
[3] X. Chen, Z. He, and L. Sun, "A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation," Transportation Research Part C: Emerging Technologies, vol. 98, pp. 73–8.

Motivations

- ITD is structured as a third-order tensor with OD pairs, time slots, and dates as the three axes

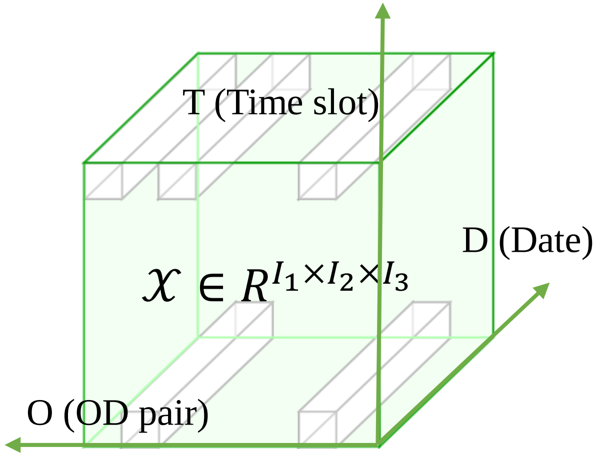

Fig1. 3-way internet traffic tensor

- There are three major open questions yet to be answered for ITDI:
  - the global low rankness of ITD has not been well defined
  - the local spatial-temporal information should not be neglected
  - efficient optimization algorithm is needed to find the stationary points and attain optimal solutions

⟹ Low-rank Spatiotemporal Priors enhanced Tucker Decomposition (LSPTD) model

Related works

- Low-rank property, which depicts the inherent global information in ITD, is an essential assumption in the TDI problem, such as

1. Nuclear norm minimization [4]
2. Tucker-based model [5]
3. CP decomposition model [6]
4. Tensor train (TT)-based model
5. tSVD-based model

- Given the spatiotemporal features in ITD, regularization terms were considered, such as
  - Chen et al. [7] proposed a tensor nuclear norm minimization considering **temporal** variation（**LATC**）
  - Wu et al. [8] proposed a modified CP decomposition using **regularizations** （**FCP**）
  - Zhang et al. [9] introduced TT rank with **temporal** regularization （**TT**）
  - Pan et al. [10] proposed an enhanced Tucker model considering **temporal** regularization （**LR-SETD**）

[4] H. Tan, G. Feng, J. Feng, W. Wang, Y.-J. Zhang, and F. Li, “A tensor-based method for missing traffic data completion,” Transportation Research Part C: Emerging Technologies, vol. 28, pp. 15–27, 2013.

[5] H. Tan, J. Feng, Z. Chen, F. Yang, and W. Wang, “Low multilinear rank approximation of tensors and application in missing traffic data,” Advances in Mechanical Engineering, vol. 6, pp. 1575–1597, 2014.

[6] K. Xie, L. Wang, X. Wang, G. Xie, J. Wen, and G. Zhang, “Accurate recovery of internet traffic data: A tensor completion approach,” in The 35th Annual IEEE International Conference on Computer Communications, 2016.

[7] X. Chen, M. Lei, N. Saunier, and L. Sun, “Low-rank autoregressive tensor completion for spatiotemporal traffic data imputation,” IEEE Transactions on Intelligent Transportation Systems, pp. 1–10, 2021.

[8] Y. Wu, H. Tan, Y. Li, J. Zhang, and X. Chen, “A fused cp factorization method for incomplete tensors,” IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 3, pp. 751–764, 2019.

[9] Z. Zhang, C. Ling, H. He, and L. Qi, “A tensor train approach for internet traffic data completion,” Annals of Operations Research, no. 6, pp. 860–889, 2021.

[10] C. Pan, C. Ling, H. He, L. Qi, and Y. Xu, “Low-rank and sparse enhanced tucker decomposition for tensor completion. arXiv 2021.

Related works

- In summary, spatial information is not fully considered

TC methods	Types of the prior		
	Low-rank	Spatial	Temporal
LR-SETD [10]	✓		✓
LATC [7]	✓		✓
TT [9]	✓		✓
FCP [8]	✓	✓	

- Gong et al. [11] proposed an innovative manifold regularized Tucker decomposition model for traffic data imputation, which considers spatial and temporal information

$$\underset{\mathcal{G};\{\mathbf{U}_n\};\mathcal{X}}{\text{minimize}}\ \mathbb{F}(\mathcal{G},\{\mathbf{U}_n\},\mathcal{X})$$
$$\triangleq \{\frac{1}{2}\|\mathcal{X}-\mathcal{G}\times_{n=1}^{N}\mathbf{U}_n\|_F^2+\beta\|\mathcal{G}\|_1$$
$$\boxed{+\frac{\alpha}{2}(\sum_{n=1}^{j}\operatorname{tr}(\mathbf{U}_n^{\mathrm{T}}\mathbf{L}_n\mathbf{U}_n)+\sum_{n=j+1}^{N}\|\mathbf{T}\mathbf{U}_n\|_F^2)\}}$$
$$s.t.\ \mathbf{U}_n\in\mathbb{R}_+^{I_n\times I_n}, n=1,\ldots,N \text{ and } \mathcal{X}_\Omega=\mathcal{X}_\Omega^0,$$

- However, the global low rankness in the Tucker model has not been well defined (considered only core tensor sparsity, but not nuclear norm of factor matrices that has a better performance)

[11] W. Gong, Z. Huang, and L. Yang, "Manifold regularized tucker decomposition for spatiotemporal traffic data imputation. arXiv 2023.

Framework

- Low-rank Spatiotemporal Priors enhanced Tucker Decomposition (LSPTD) model

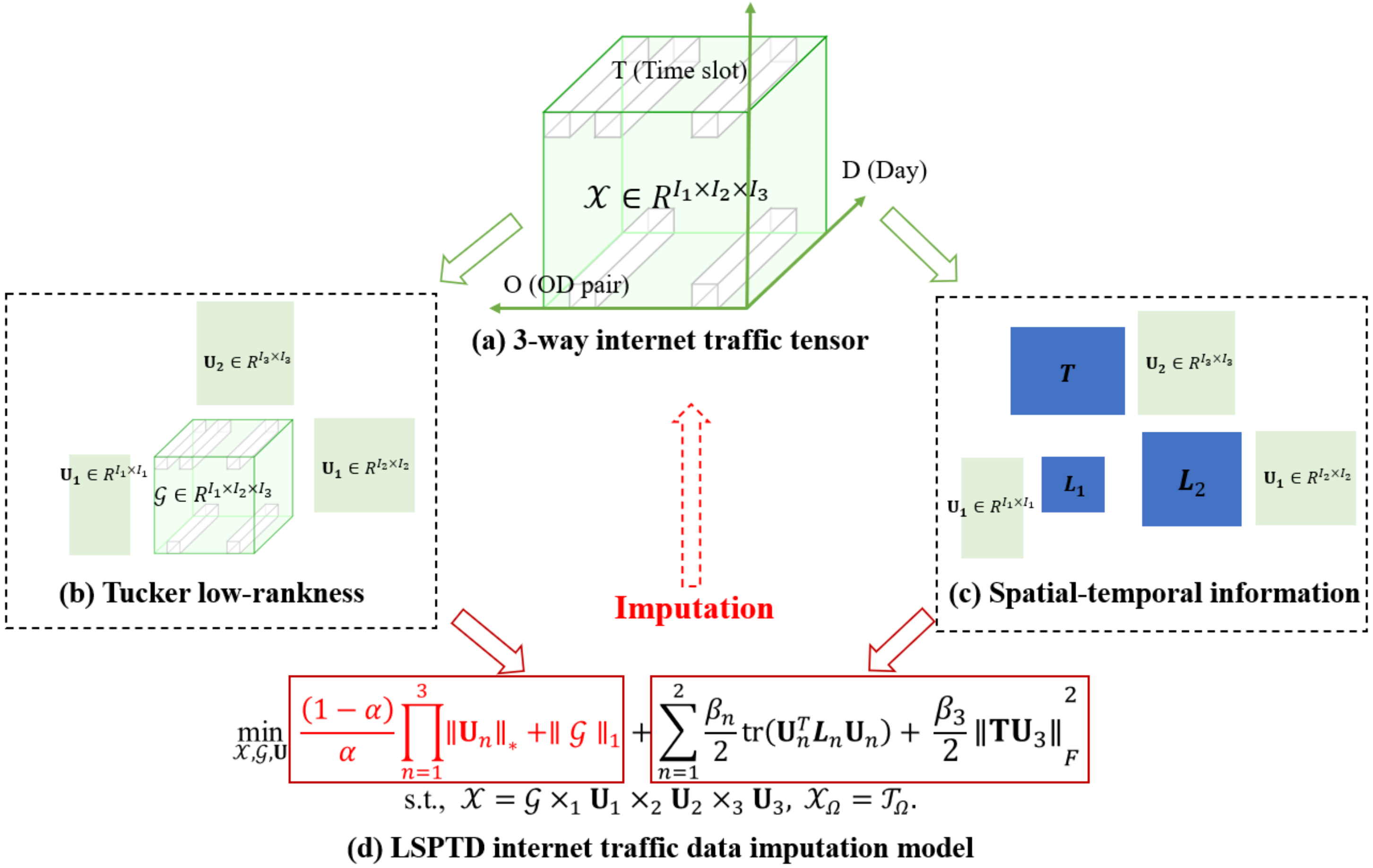

Fig 2. Visual display for the proposed LSPTD model

Proposed model: LSPTD

- we use the Tucker decomposition to characterize the global low-rank prior

$$\min_{\mathcal{G},\mathbf{U}_n} (1-\alpha)\prod_{n=1}^{N}\|\mathbf{U}_n\|_* + \alpha \| \mathcal{G} \|_1$$
$$\text{s.t. } \mathcal{X} = \mathcal{G} \mathop{\times}_{n=1}^{N} \mathbf{U}_n,\ \ 0<\alpha<1,$$

- we use the factor GE prior and the Toeplitz matrix as smoothness constraints to characterize the spatiotemporal correlations

$$\min_{\mathcal{G},\mathbf{U}_n} \frac{(1-\alpha)}{\alpha}\prod_{n=1}^{N}\|\mathbf{U}_n\|_* + \| \mathcal{G} \|_1 + \sum_{n=1}^{N-1}\frac{\beta_n}{2}\mathrm{tr}\left(\mathbf{U}_n^{\mathrm{T}}\mathbf{L}_n\mathbf{U}_n\right) + \frac{\beta_N}{2}\|\mathbf{T}\mathbf{U}_N\|_F^2$$
$$\text{s.t.} \quad \mathcal{X} = \mathcal{G} \mathop{\times}_{n=1}^{N} \mathbf{U}_n,$$
$$0<\alpha<1,\quad \beta_n \geq 0,\ n=1,2,\cdots,N.$$

Remark: the product function of the LSPTD model is nonconvex, which is hard to solve. Also, the product only represents the block size of the core tensor, we thus use the weighted factor matrix nuclear norm summation term in our algorithm design.

Proposed model: LSPTD

$$
\begin{aligned}
\min_{\mathcal{G},\{\mathbf{U}_n\}} \quad & \lambda \sum_{n=1}^{3} \omega_n \|\mathbf{U}_n\|_* + \|\mathcal{G}\|_1 \\
& + \sum_{n=1}^{2} \frac{\beta_n}{2} \operatorname{tr}(\mathbf{U}_n^{\mathrm{T}} \mathbf{L}_n \mathbf{U}_n) + \frac{\beta_3}{2} \|\mathbf{T}\mathbf{U}_3\|_F^2, \\
\text{s.t.} \quad & \mathcal{X} = \mathcal{G} \mathop{\times}_{n=1}^{3} \mathbf{U}_n, \quad \mathcal{X}_\Omega = \mathcal{T}_\Omega,
\end{aligned}
$$

where

$$\lambda = \frac{1-\alpha}{\alpha},\ 0 < \alpha < 1,\ \omega_n = \prod_{i=1, i\neq n}^{3} \frac{1}{R_i},\ R_i = \sum \sigma(\mathbf{U}_i).$$

Remark: LSPTD is a nonconvex minimization problem (mode-n is a bilinear function) containing many local minima; it is much harder to find the optimal solution.

Two implementable algorithms

- **Algorithm 1**: Proximal Alternating Linearized Minimization

$$
\begin{aligned}
\min_{\mathcal{G},\{\mathbf{U}_n\},\mathcal{X}} \quad & \lambda \sum_{n=1}^{3} \omega_n \|\mathbf{U}_n\|_* + \|\mathcal{G}\|_1 + \frac{\mu}{2}\left\|\mathcal{X} - \mathcal{G} \times_{n=1}^{3} \mathbf{U}_n\right\|_F^2 \\
& + \sum_{n=1}^{2} \frac{\beta_n}{2} \operatorname{tr}(\mathbf{U}_n^T \mathbf{L}_n \mathbf{U}_n) + \frac{\beta_3}{2}\|\mathbf{T}\mathbf{U}_3\|_F^2, \\
\text{s.t.} \quad & \mathcal{X}_\Omega = \mathcal{T}_\Omega, \quad \mathcal{X}_{\bar{\Omega}} = \{\mathcal{G} \times_{n=1}^{3} \mathbf{U}_n\}_{\bar{\Omega}}.
\end{aligned}
$$

$$
\hat{\mathcal{G}} = \operatorname*{argmin}_{\mathcal{G}} \frac{\mu L_{\mathcal{G}}}{2}\left\|\mathcal{G} - \left(\tilde{\mathcal{G}} - \frac{1}{L_{\mathcal{G}}}\nabla_{\mathcal{G}} f(\tilde{\mathcal{G}})\right)\right\|_F^2 + \|\mathcal{G}\|_1
$$

$$
\nabla_{\mathcal{G}} f(\mathcal{G}) = \mathcal{G} \times_{n=1}^{3} \mathbf{U}_n^T\mathbf{U}_n - \mathcal{X} \times_{n=1}^{3} \mathbf{U}_n^T \qquad L_{\mathcal{G}} = \|\otimes_{n=3}^{1} \mathbf{U}_n^T\mathbf{U}_n\|_2 = \prod_{n=1}^{3}\|\mathbf{U}_n^T\mathbf{U}_n\|_2
$$

$$
\begin{aligned}
\hat{\mathbf{U}}_n = \; & \operatorname*{argmin}_{\mathbf{U}_n} \frac{L_{\mathbf{U}_n}}{2}\left\|\mathbf{U}_n - \left(\tilde{\mathbf{U}}_n - \frac{1}{L_{\mathbf{U}_n}}\nabla_{\mathbf{U}_n} f(\tilde{\mathbf{U}}_n)\right)\right\|_F^2 \\
& + \frac{\lambda\omega_n}{\mu}\|\mathbf{U}_n\|_*.
\end{aligned}
$$

$$
\begin{aligned}
\nabla_{\mathbf{U}_n} f(\mathbf{U}_n) \; & = \mathbf{U}_n \mathbf{G}_{(n)} \mathbf{V}_n^T \mathbf{V} \mathbf{G}_{(n)}^T - \mathbf{X}_{(n)} \mathbf{V} \mathbf{G}_{(n)}^T \\
& + \beta_n \mathbf{L}_n \mathbf{U}_n
\end{aligned}
$$

$$
L_{\mathbf{U}_n} = \|\mathbf{G}_{(n)} \mathbf{V}_n^T \mathbf{V} \mathbf{G}_{(n)}^T\|_2 + \|\beta_n \mathbf{L}_n\|_2
$$

Algorithm 1 PLAM-based LSPTD

Input: Missing traffic tensor $\mathcal{T}$, observed entries Ω.
Output: Imputed traffic tensor $\hat{\mathcal{X}}$.
Initialize $\mathcal{G}^0, \{\mathbf{U}_n^0\}$ $(1 \le n \le 3)$ randomly, $0 < \alpha < 1$, $\mu = 1$, and define $\mathcal{Z}^0$ as null tensor;
$\mathcal{X}_\Omega = \mathcal{T}_\Omega$, $\mathcal{X}_{\bar{\Omega}} = \mathcal{Z}_{\bar{\Omega}}^0$;
while $k < K$ **do**
  Optimize $\mathcal{G}$ by Eq. (6);
  for $n = 1$ to 3 **do**
    Optimize $\mathbf{U}_n$ using Eq. (10);
  end for
  Update Tucker decomposition $\mathcal{Z}^k$ using $\{\mathbf{U}_n^k\}$ and $\mathcal{G}^k$;
  if $\mathbb{F}\left(\mathcal{G}^k, \mathbf{U}_{j \le n}, \mathbf{U}_{j > n}\right)$ is decreasing **then**
    Re-update $\mathcal{G}^k$ and $\mathbf{U}_n^k$ respectively;
  else
    Re-update $\mathcal{G}^k$ and $\mathbf{U}_n^k$ respectively with $\tilde{\mathcal{G}}^k = \mathcal{G}^{k-1}$ and $\tilde{\mathbf{U}}_n^k = \mathbf{U}_n^{k-1}$;
  end if
  until $\left\|\mathcal{X}^{k+1} - \mathcal{X}^k\right\|_F \left\|\mathcal{X}^k\right\|_F^{-1} < 1e^{-3}$ are satisfied.
end while
return $\hat{\mathcal{X}}_\Omega = \mathcal{T}_\Omega$, $\hat{\mathcal{X}}_{\bar{\Omega}} = \left(\hat{\mathcal{G}} \times_{n=1}^{3} \hat{\mathbf{U}}_n\right)_{\bar{\Omega}}$.

Two implementable algorithms

- **Algorithm 2**: Inexact Augmented Lagrange Multiplier Framework

$$\begin{aligned}\min_{\mathcal{G},\mathbf{U}_n,\mathcal{X}} \quad & \lambda\sum_{n=1}^{3}\omega_n\|\mathbf{U}_n\|_* + \|\mathcal{G}\|_1 \\ & + \sum_{n=1}^{2}\frac{\beta_n}{2}\operatorname{tr}(\mathbf{U}_n^{\mathrm{T}}\mathbf{L}_n\mathbf{U}_n) + \frac{\beta_3}{2}\|\mathbf{T}\mathbf{U}_3\|_F^2 \\ & + \frac{\mu}{2}\left\|\mathcal{X}-\mathcal{G}\mathop{\times}_{n=1}^{3}\mathbf{U}_n\right\|_{\mathrm{F}}^2 + \left\langle \mathcal{Y}, \mathcal{X}-\mathcal{G}\mathop{\times}_{n=1}^{3}\mathbf{U}_n\right\rangle\end{aligned}$$

$\Rightarrow$

$$\hat{\mathcal{G}} = \mathcal{S}_{\frac{1}{\mu L_{\mathcal{G}}}}\left(\tilde{\mathcal{G}} - \frac{1}{L_{\mathcal{G}}}\nabla_{\mathcal{G}} f(\tilde{\mathcal{G}})\right),\ L_{\mathcal{G}} = \prod_{n=1}^{3}\|\mathbf{U}_n^{\mathrm{T}}\mathbf{U}_n\|_2$$

$$\nabla_{\mathcal{G}} f(\mathcal{G}) = \mathcal{G}\mathop{\times}_{n=1}^{3}\mathbf{U}_n^{\mathrm{T}}\mathbf{U}_n - \left(\mathcal{X}+\frac{\mathcal{Y}}{\mu}\right)\mathop{\times}_{n=1}^{3}\mathbf{U}_n^{\mathrm{T}}.$$

$$\begin{aligned}\hat{\mathbf{U}}_n &= \mathcal{D}_{\frac{\lambda\omega_n}{L_{\mathbf{U}_n}}}\left(\tilde{\mathbf{U}}_n - \frac{1}{L_{\mathbf{U}_n}}\nabla_{\mathbf{U}_n} f(\tilde{\mathbf{U}}_n)\right). \\ \nabla_{\mathbf{U}_n} f(\mathbf{U}_n) &= \mu\mathbf{U}_n\mathbf{G}_{(n)}\mathbf{V}_n^{\mathrm{T}}\mathbf{V}\mathbf{G}_{(n)}^{\mathrm{T}} + \beta_n\mathbf{L}_n\mathbf{U}_n \\ &\quad -(\mu\mathbf{X}_{(n)} + \mathbf{Y}_{(n)})\mathbf{V}\mathbf{G}_{(n)}^{\mathrm{T}}, \\ L_{\mathbf{U}_n} &= \left\|\mu\mathbf{G}_{(n)}\mathbf{V}_n^{\mathrm{T}}\mathbf{V}\mathbf{G}_{(n)}^{\mathrm{T}}\right\|_2 + \|\beta_n\mathbf{L}_n\|_2\end{aligned}$$

Algorithm 2 IALM-based LSPTD

1: **Input**: Missing traffic tensor $\mathcal{T}$, observed entries Ω.
2: **Output**: Imputed traffic tensor $\hat{\mathcal{X}}$.
3: **Initialize**: $\mathcal{G}^0, \{\mathbf{U}_n^0\}$ $(1 \le n \le 3)$, $0 < \alpha < 1$, $\mu^0 = 1e^{-5}$, and define $\mathcal{Z}^0$ as null tensor;
4: $\mathcal{X}_\Omega = \mathcal{T}_\Omega$, $\mathcal{X}_{\bar{\Omega}} = \mathcal{Z}_{\bar{\Omega}}^0$;
5: **while** $k < K$ **do**
6: Optimize $\mathcal{G}^{k+1}$ via (14) with other variables fixed;
7: Optimize all $\mathbf{U}_n^{k+1}$ via (15) with other variables fixed;
8: Optimize $\mathcal{X}^{k+1}$ with other variables fixed;
9: Update $\mathcal{Y}^{k+1}$ and $\mu^{k+1} = \rho\mu^k$, $\rho \in [1.1, 1.2]$
10: **until** $\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F \|\mathcal{X}^k\|_F^{-1} < 1e^{-3}$ are satisfied.
11: **end while**
12: **return** $\hat{\mathcal{X}}_\Omega = \mathcal{T}_\Omega$, $\hat{\mathcal{X}}_{\bar{\Omega}} = \left(\hat{\mathcal{G}} \times_{n=1}^3 \hat{\mathbf{U}}_n - \frac{\mathcal{Y}_K}{\mu_K}\right)_{\bar{\Omega}}$

Numerical results

- Datasets

We employed the **Abilene** dataset and the **GÉANT** dataset, obtained the numerical results.

https://doi.org/10.5281/zenodo.7725126   https://totem.info.ucl.ac.be/dataset.html

- Performance evaluation

We adopt the normalized mean absolute error (NMAE, the lower the better) value to measure the performance

$$\mathrm{NMAE} = \frac{\sum_{(i_1,i_2,i_3)\in\bar{\Omega}} \left|\widehat{\mathcal{X}}_{i_1 i_2 i_3} - \mathcal{X}^*_{i_1 i_2 i_3}\right|}{\sum_{(i_1,i_2,i_3)\in\bar{\Omega}} \left|\mathcal{X}^*_{i_1 i_2 i_3}\right|},$$

where $\widehat{\mathcal{X}}$ and $\mathcal{X}^*$ correspond to the imputed and real tensor, respectively.

- Implementation details

Our experiments consider:

- random missing (RM) with the sample ratio (SR) from 0.9 to 0.05
- structurally missing (SM) scenarios

In all our experiments, we set $\alpha = 0.5$ and calculate the SVD ratios between mode-n unfolding matrices and spatiotemporal matrices ($\{\mathbf{L}_1, \mathbf{L}_2\}$ and $\mathbf{T}$ to deliver β_n, $n = 1,2,3$.

Model comparison

The red lines represents the results from LSPTD model

- Baselines: ManiRTD , LR-SETD , LATC , and TAS-LR

  It is noteworthy that TAS-LR is a direct matrix-based approach, while others are tensor-based.

- RM scenarios

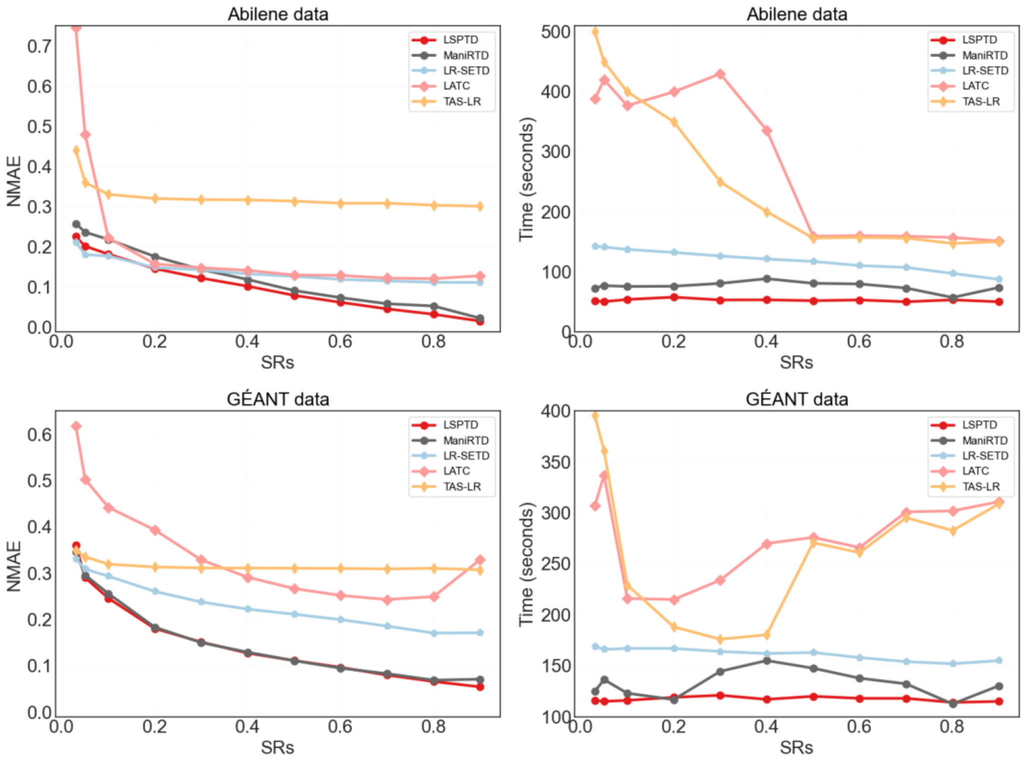

Fig 3. RM results of ITDI methods based on NMAE values (left) and computing time (right) across different SRs for both the Abilene dataset (upper) and the GÉANT dataset (lower).

- SM scenarios

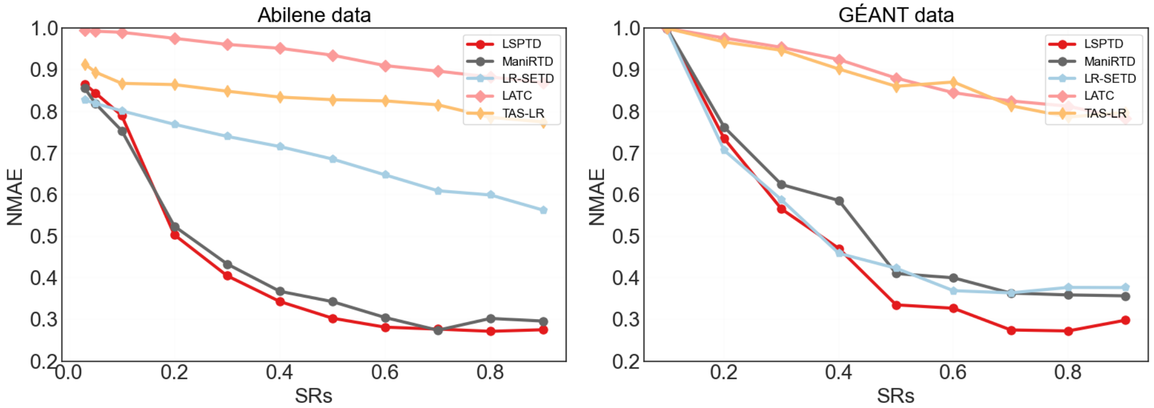

Fig 4. SM results of ITDI methods based on NMAE values across different SRs for both the Abilene dataset (upper) and the GÉANT dataset (lower).

Ablation study for Abilene data

- We first discuss the effect of the low-rank, spatial, and temporal priors of LSPTD

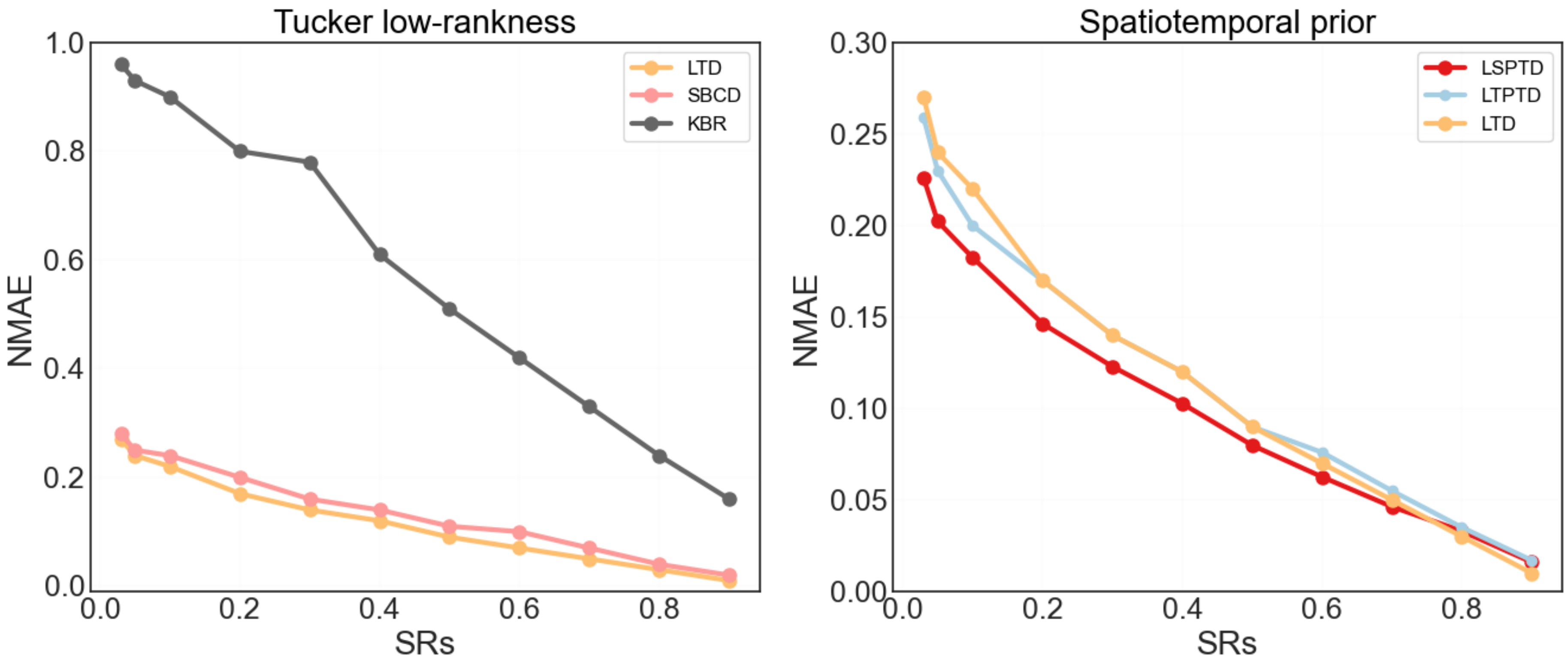

Fig 5. NMAE versus SRs for KBR, SBCD, LTD, LTPTD, and LSPTD

Algorithm comparison for Abilene data

- We investigate the performance of proposed **Algorithms 1 and 2**

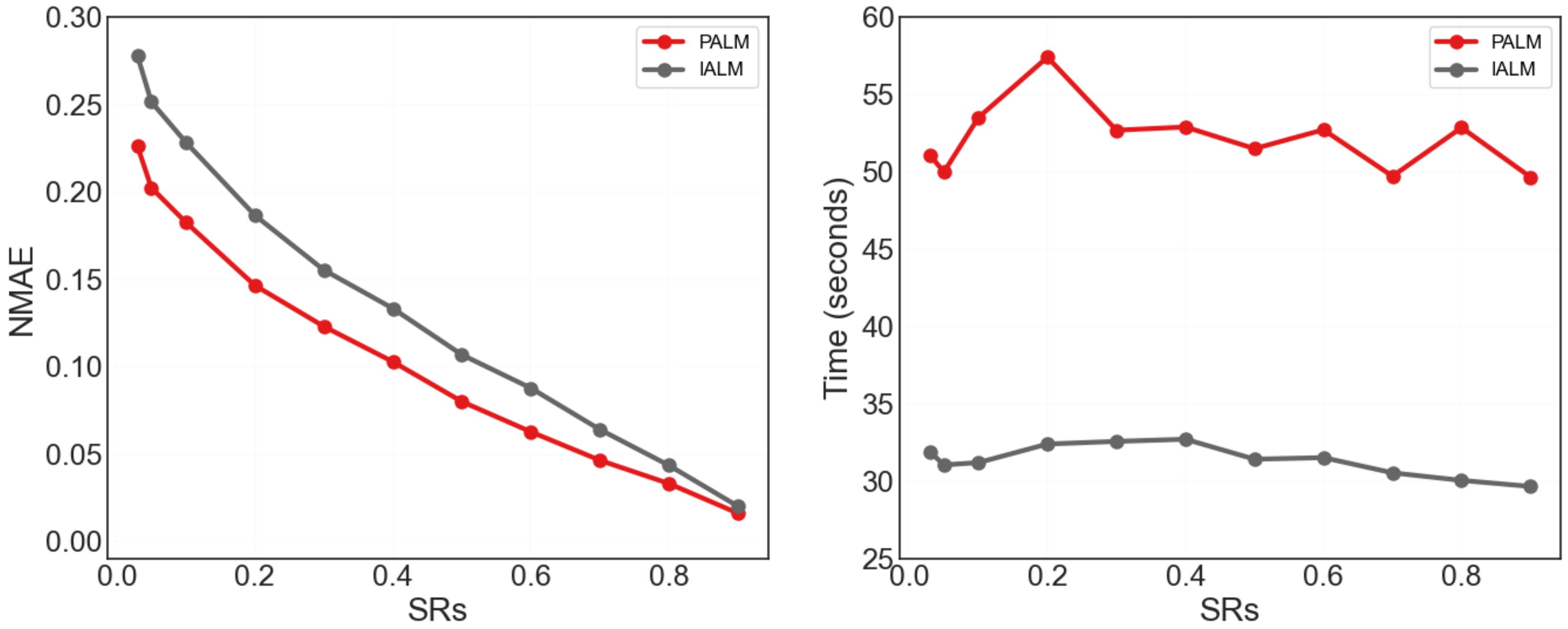

Fig 6. Results of NMAE and computing time versus SRs via solving LSPTD model using Algorithm 1 and Algorithm 2.

Conclusions

- We propose a novel Tucker-based model with integrated low-rank and spatiotemporal priors (LSPTD) for internet traffic data imputation.

- The proposed LSPTD model utilizes weighted factor matrix rank and core tensor sparsity to capture global low rankness. Additionally, it incorporates the factor GE and a Toeplitz matrix as spatiotemporal constraints to enhance model performance.

- Two optimization algorithms are specifically designed.

- A series of experiments validate the superiority of our proposal, showing its superior accuracy and efficiency compared to existing matrix-based and tensor-based methods.