Temporal Graph Regression via Structure-Aware Intrinsic Representation Learning

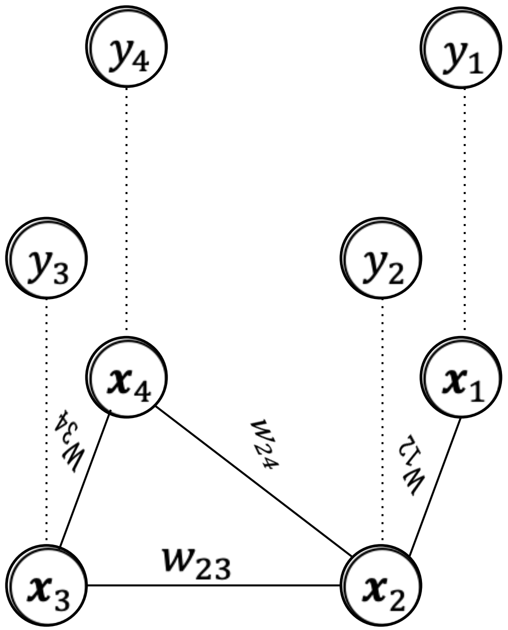
Chao Han., Xi Hang Cao, Marija Stanojevic, Mohamed Ghalwash,

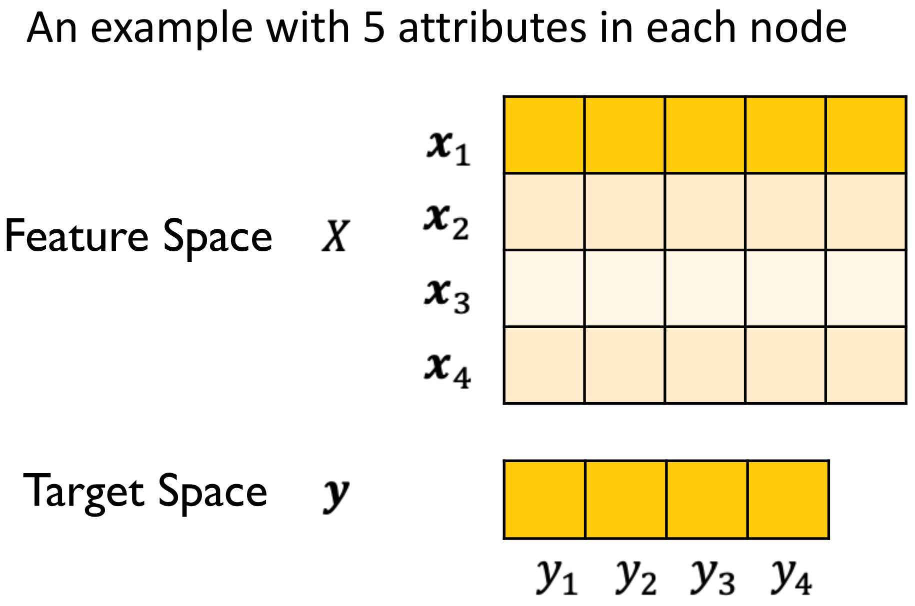
Zoran Obradovic (Presenter)

SDM 2019

Attributed Graph

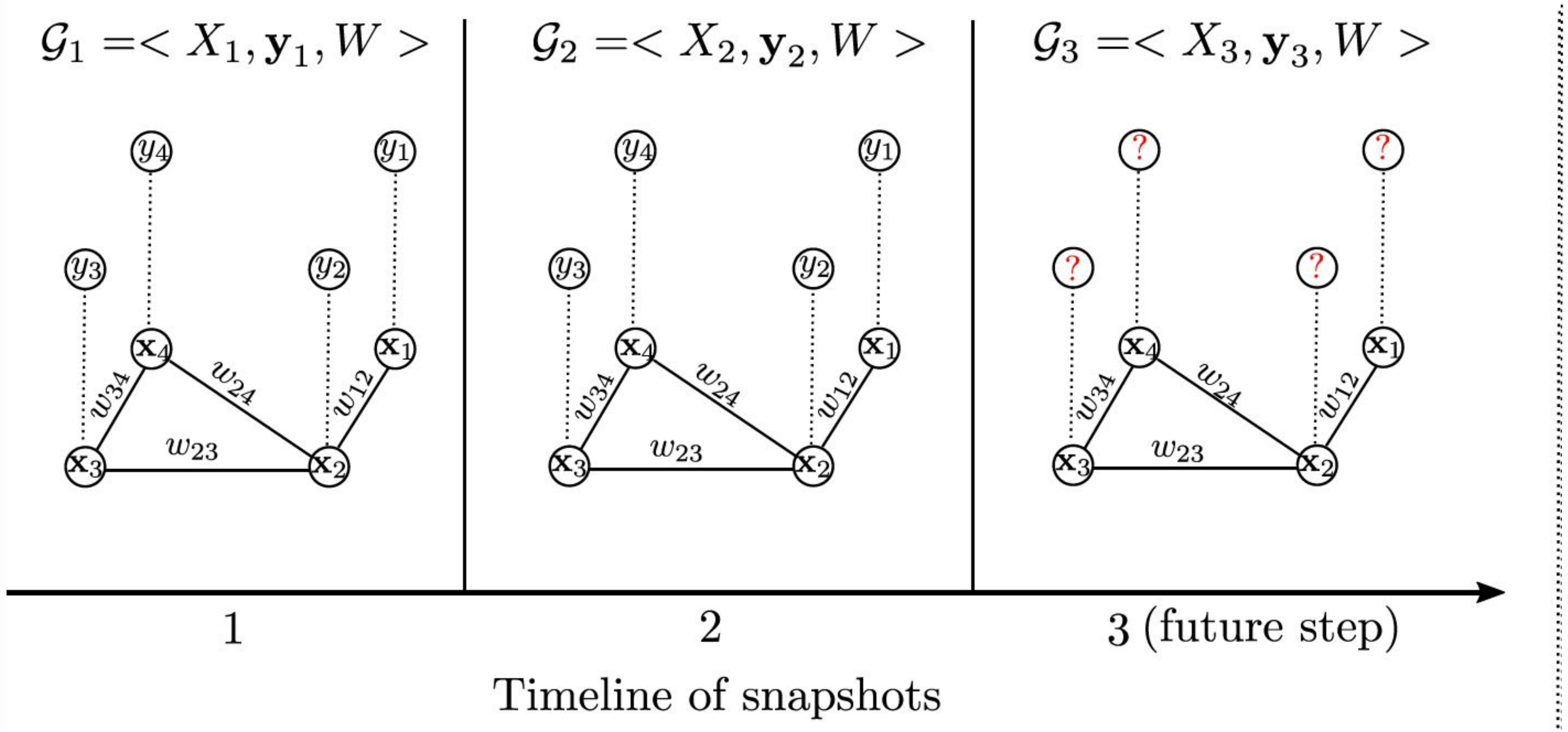
- Node i is composed of a target variable and a vector of attributes. $\{y_i, \boldsymbol{x}_i\}$
- The edge between node i and node j (w_{ij}) is determined by prior knowledge, or similarity, or specific algorithmically calculations between nodes.



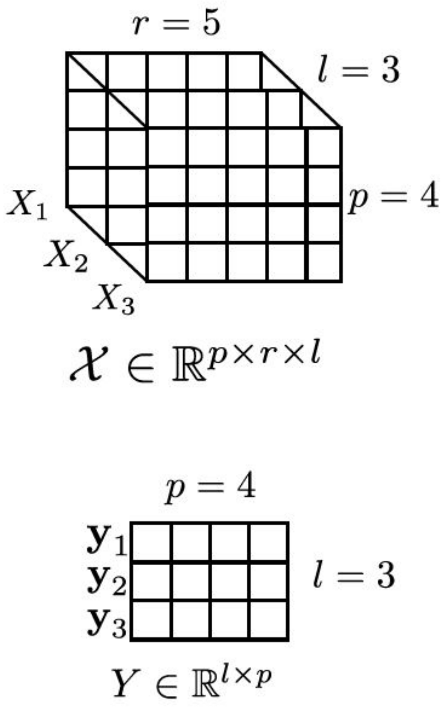


Temporal Graph Regression

Goal: Predict target variables y_i at future time step



Matrix view of the temporal graph



\# snapshots: l
\# nodes: p
\# features in node: r

Problems from the graph representation

- Number of variables in the input space: $O(l * p * r)$
- Number of variables in the target space: $O(l * p)$
- **Problems:**
  - Information contained in the data is redundant
  - Model complexity is proportional to the number of variables
- **Task:** Find good representation of the temporal graph, such that the temporal graph regression benefits from it
- **Solution:** Learn jointly latent feature space and latent target space by considering the uniqueness of temporal graph.

Related work

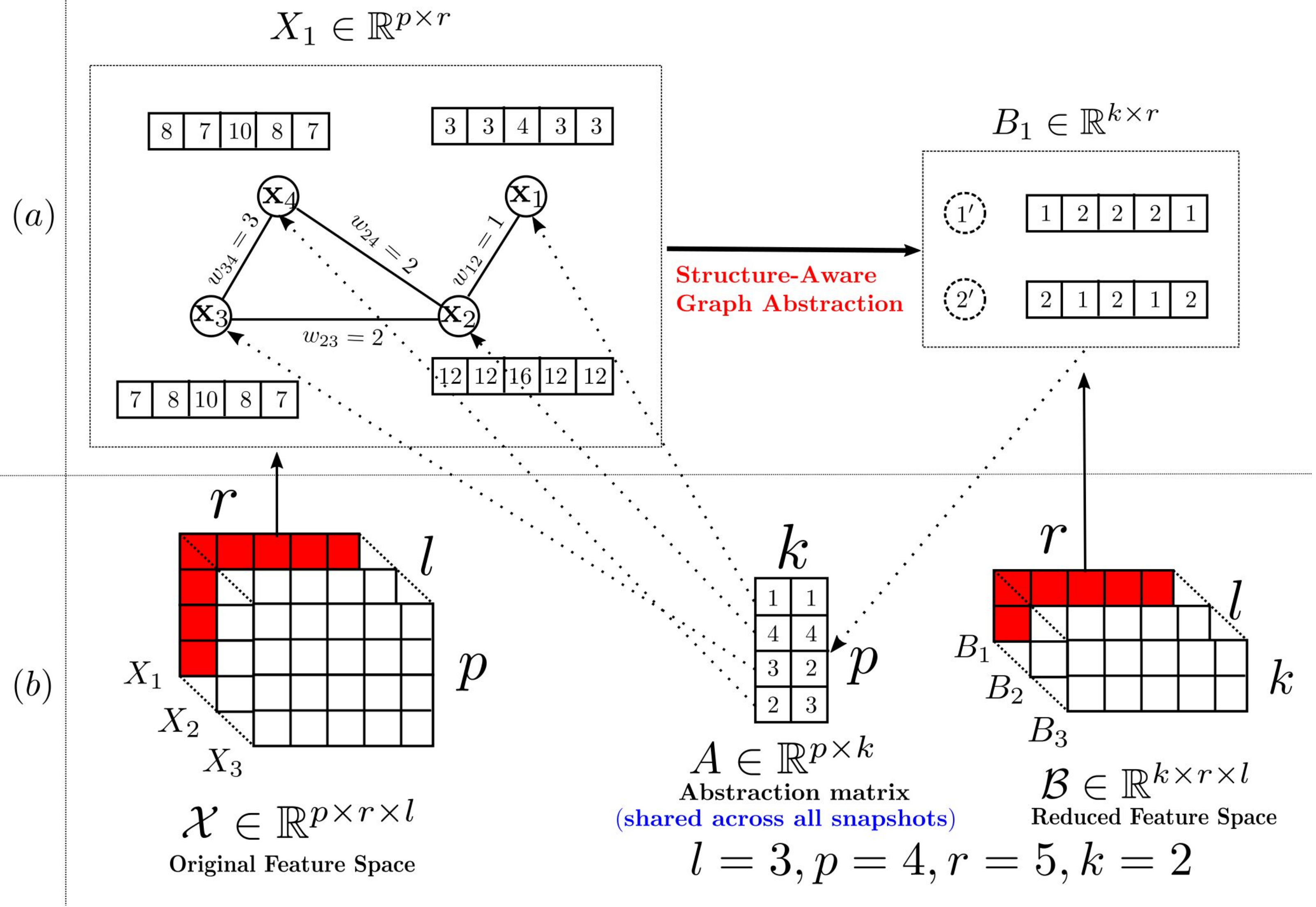
- Many traditional approaches focused on learning low-dimensional representation of the feature spaces, but not for temporal graph data.
- Some recent works applied dimension reduction techniques on the target space. [PLST'12, CPLST'14, FAIE'14].
- None of them consider the representation learning on both spaces.

Structure-aware Graph Abstraction

Graph Abstraction:
representing graph with fewer nodes (p nodes are summarized into k nodes, k < p)

$$\min_{\mathcal{B},A} \left\| \mathcal{X} - \mathcal{B} \times_1 A \right\|_F^2$$

$\mathcal{X}$: feature space tensor
$\mathcal{B}$: latent feature space tensor
A: graph abstraction matrix



Structure-aware Graph Abstraction (Cont.)

- **Temporal Smoothness:** neighboring graphs on timeline are similar

$$\min_{\mathrm{B}} \sum_{i=1}^{l-1} ||B_i - B_{i+1}||_2^2$$

- **Graph Structure Preservation:** if two nodes are close then their abstractions should also be similar

$$\min_{\mathrm{A}} tr(A^T L A)$$

L is the Laplacian matrix of the similarity matrix W.

Feature-Aware Target Space Learning

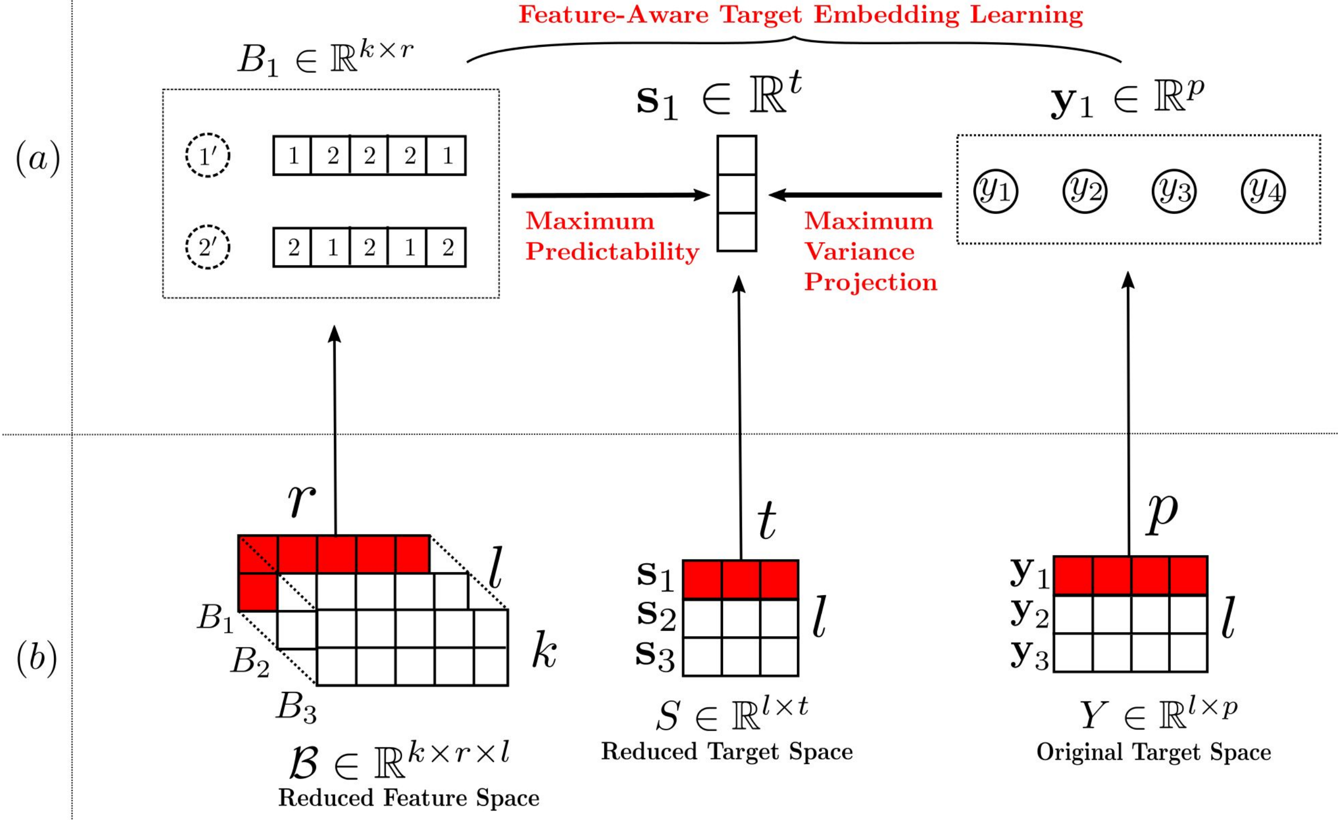
Maximum Predictability: maintain the predictability of the latent target space ***s***

$$\min_{U,V} ||YV - \mathcal{B}_{(3)}U||_F^2$$

$\mathcal{B}_{(3)}$ is the concatenation of the latent feature matrices in each snapshot. i.e., $\mathcal{B}_{(3)} = [B_1(:), \cdots, B_l(:)]$

Maximum Variance Projection: find a projection such that the reconstruction error is minimized

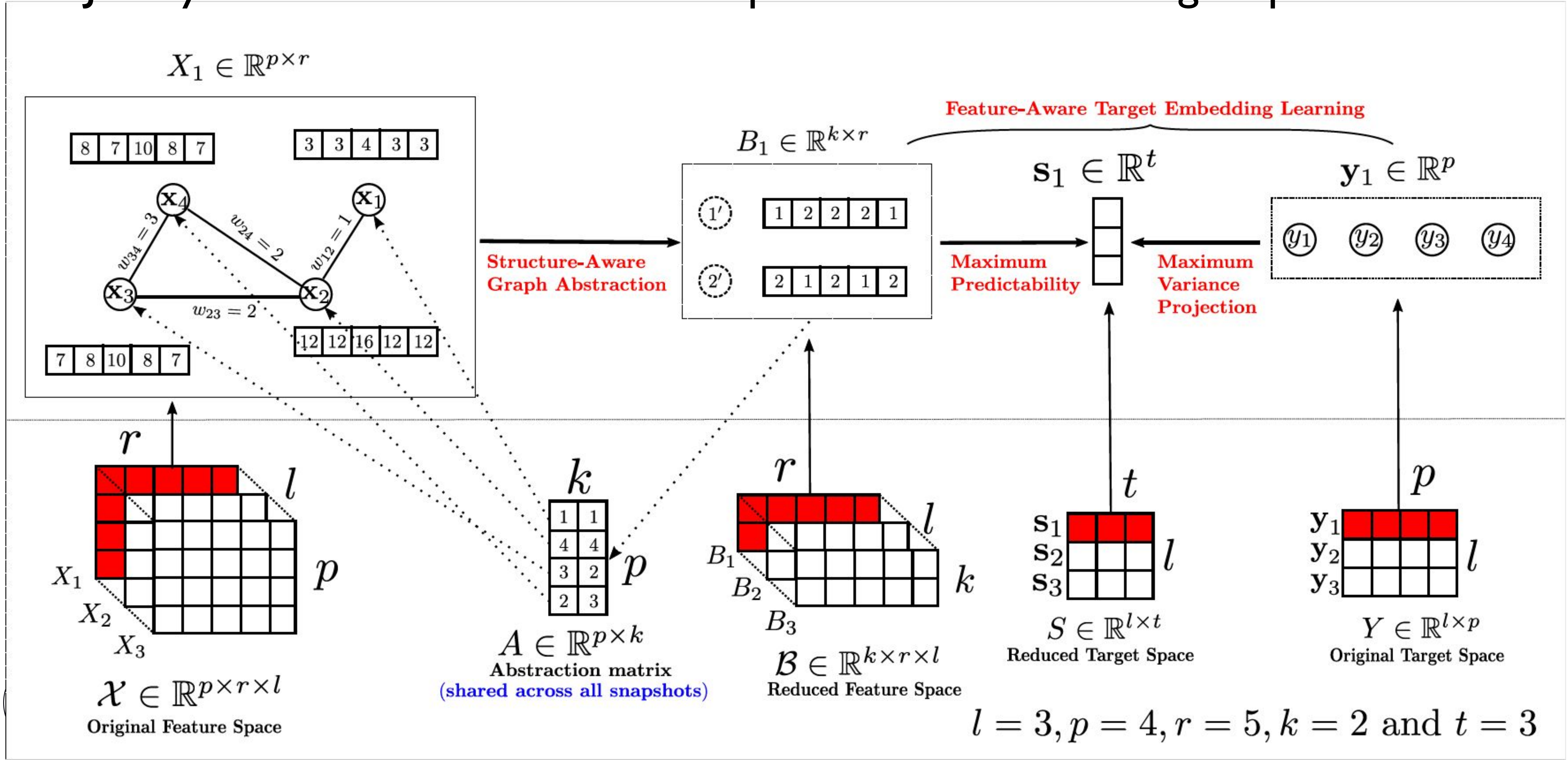
$$max_{V^TV=I}\, tr(V^TYYV)$$



$l = 3, p = 4, r = 5, k = 2$ and $t = 3$

Intuition

Jointly discover the latent feature space and the latent target space.



The joint learning problem

$$f = \underbrace{||\mathcal{X} - \mathcal{B} \times_1 A||_F^2}_{\text{Shared Abstraction}} + \delta \sum_{i=1}^{l-1} \underbrace{||B_i - B_{i+1}||_2^2}_{\text{Temporal Smoothness}}$$
$$+ \underbrace{||\mathcal{B}_{(3)} U - YV||_F^2}_{\text{Maximum Predictability}} - \underbrace{tr(V^T Y^T Y V)}_{\text{Maximum Variance}} + \underbrace{\alpha tr(A^T L A)}_{\text{Structure Preservation}}$$

$$\{A^*, \mathcal{B}^*, U^*, V^*\} = argmin_{A,\mathcal{B},U,V^T V = I} f$$

Derivative-free block coordinate descent algorithm is proposed to solve it
All sub-problems have closed-form solution.

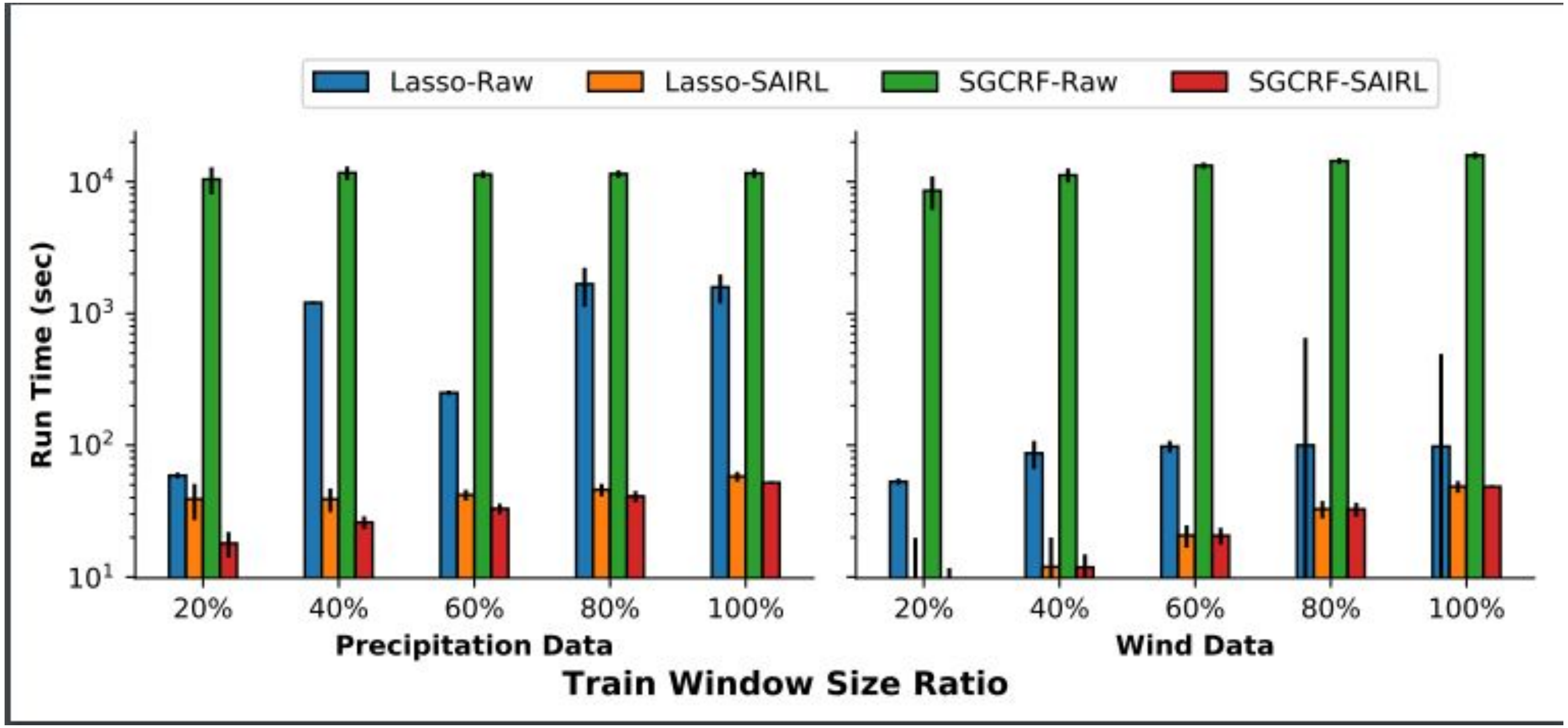
Datasets

- **Precipitation Forecasting**
  - Collected from 124 U.S. cities in 708 snapshots in monthly resolution. (l=708 snapshots, p = 124 nodes, r = 9 features per node).
  - **Task:** forecast precipitation in the next month at all locations.
  - The similarity W is calculated as the inverse distance between two locations.
- **Wind Forecasting**
  - Collected from 7 wind farms with 4 features in each over 1080 days. (l=1080 snapshots, p = 168 nodes and r = 4 features per node)
  - **Task:** predict hourly wind power of all 7 farms in the next day.
  - Similarity w_{ij} is 1 if node i and node j are within the same hour or they correspond to the same node of neighboring hours and 0 otherwise.

Results

- Compared the proposed representation learning method with four methods: raw, CPLST, FaIE, SAGA.
- Evaluated the quality embedding using two regression methods for temporal graph regression: Lasso and SGCRF.
- Varied the training sizes from {20%, 40%, 60%, 80%, 100%} of all training data and experimented on 8 windows for each training size.
- The embedding generated from the proposed method always lead to better MSE across all experimental settings.

Efficiency brought by the graph abstraction



Conclusion

- **Task:** find representation of the temporal graph, such that the temporal graph regression is faster and more accurate
- Proposed a joint representation learning method for temporal graph regression by utilizing the structure of temporal attributed graph
- Developed a block coordinate descent method for solving the optimization problem. All sub-problems have closed-form solutions.
- Demonstrated the effectiveness of embedding by conducting extensive experiments on two real-world datasets.

Thanks!
Q&A

Thanks!
Q&A