Spatio-Temporal Graph Few-Shot Learning with Cross-City Knowledge Transfer

Bin Lu[1], Xiaoying Gan[1], Weinan Zhang[1], Huaxiu Yao[2], Luoyi Fu[1], Xinbing Wang[1]

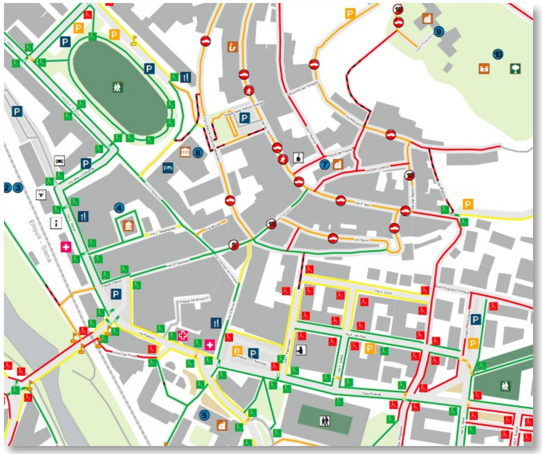
[1]Shanghai Jiao Tong University, [2]Stanford University

Table of Contents

Table of Contents

Background

- Graphs are ubiquitously used to reveal the interactions among various entities
  - Road Network/ Metro Network
  - Sensor Network/ IoT Device Network
- Spatio-temporal graph learning is a widely used method for urban computing tasks
  - Taxi Demand Prediction
  - Traffic Flow Forecasting
  - Air Quality Prediction
  - COVID-19 Cases Prediction



Background

- Existing methods[1,2,3] require large-scale training samples
  - High cost of data collection for data-scarce cities
  - Cold start scenario for urban computing applications
- Cross-city Knowledge Transfer
  - Reduce the burden of data collection
  - Improve the efficiency of smart city construction

[1] Xu et al. "Spatiotemporal Graph Convolution Multifusion Network for Urban Vehicle Emission Prediction.", TNNLS 2021.
[2] Pan et al. " Spatio-temporal meta learning for urban traffic prediction.", IEEE TKDE, 2020.
[3] Do et al. "Graph-deep-learning-based inference of fine-grained air quality from mobile IoT sensors.", IEEE IOTJ, 2020.

Existing Methods

- **Related Works about Cross-city Knowledge Transfer**[4,5,6]
  - [4] firstly proposes an one-to-one knowledge transfer
    - Face the risk of negative transfer due to the great difference between two cities.
  - [5] proposes a region-to-region matching
    - Introduce large-scale auxiliary data (social media check-ins)
  - [6] combines meta-learning with knowledge transfer
    - w/o considering varied feature across and within cities
  - Above methods are only applicable to grid-based data, but not compatible with graph-based modeling.

[4] Wei et al. “Transfer knowledge between cities. ”, KDD 2016.
[5] Wang et al. “Cross-city transfer learning for deep spatio-temporal prediction.”, IJCAI 2019.
[6] Yao et al. “Learning from multiple cities: A meta-learning approach for spatial-temporal prediction”, WWW 2019.

Problem

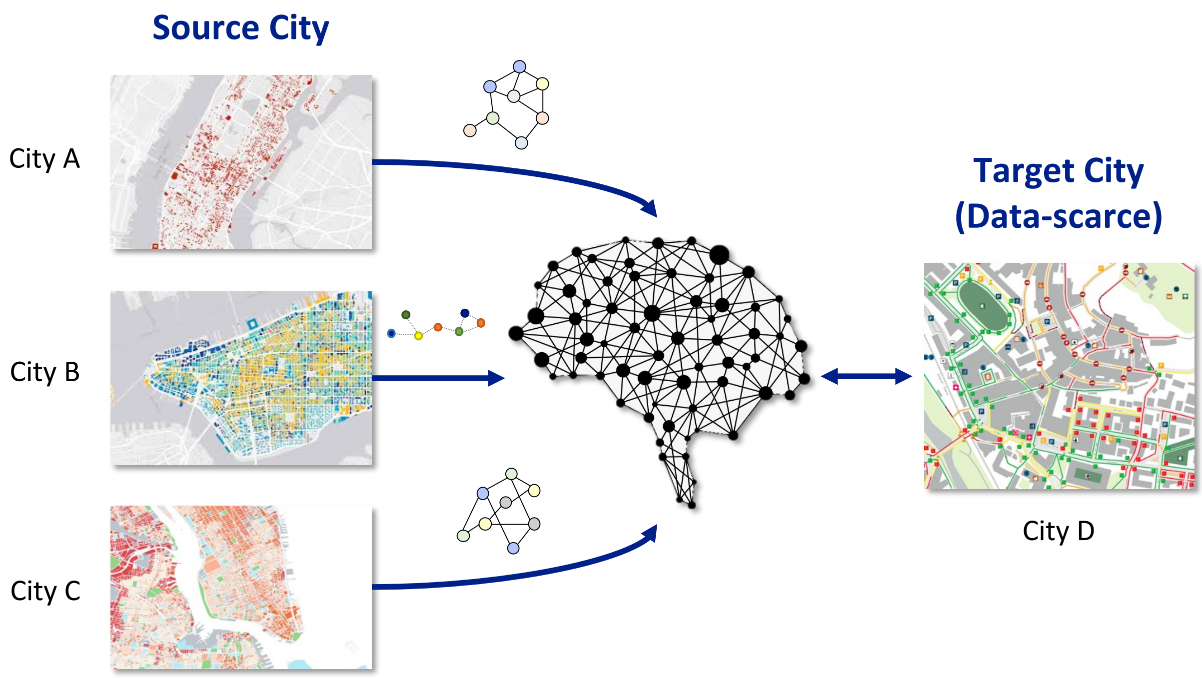
- Our goal
  - Transfer the cross-city knowledge in graph-based few-shot learning scenarios.
  - Exploring the impact of knowledge transfer across multiple cities.
- Problem Definition
  - Spatio-temporal Graph Forecasting

$$[\mathbf{X}^{t-T+1}, \cdots, \mathbf{X}^{t}; \mathcal{G}_{ST}] \xrightarrow{f(\cdot)} [\mathbf{X}^{t+1}, \cdots, \mathbf{X}^{t+M}]$$

  - Spatio-temporal Graph Few-Shot Learning
    - P source data-rich cities and 1 target data-scarce city
    - Leveraging meta knowledge learned from multiple source cities to predict on a disjoint few-shot target scenario

Challenges

- #1 How to adapt feature extraction in target city via the knowledge from multiple source cities?
- #2 How to alleviate the impacts of varied graph structure on transferring among different cities?



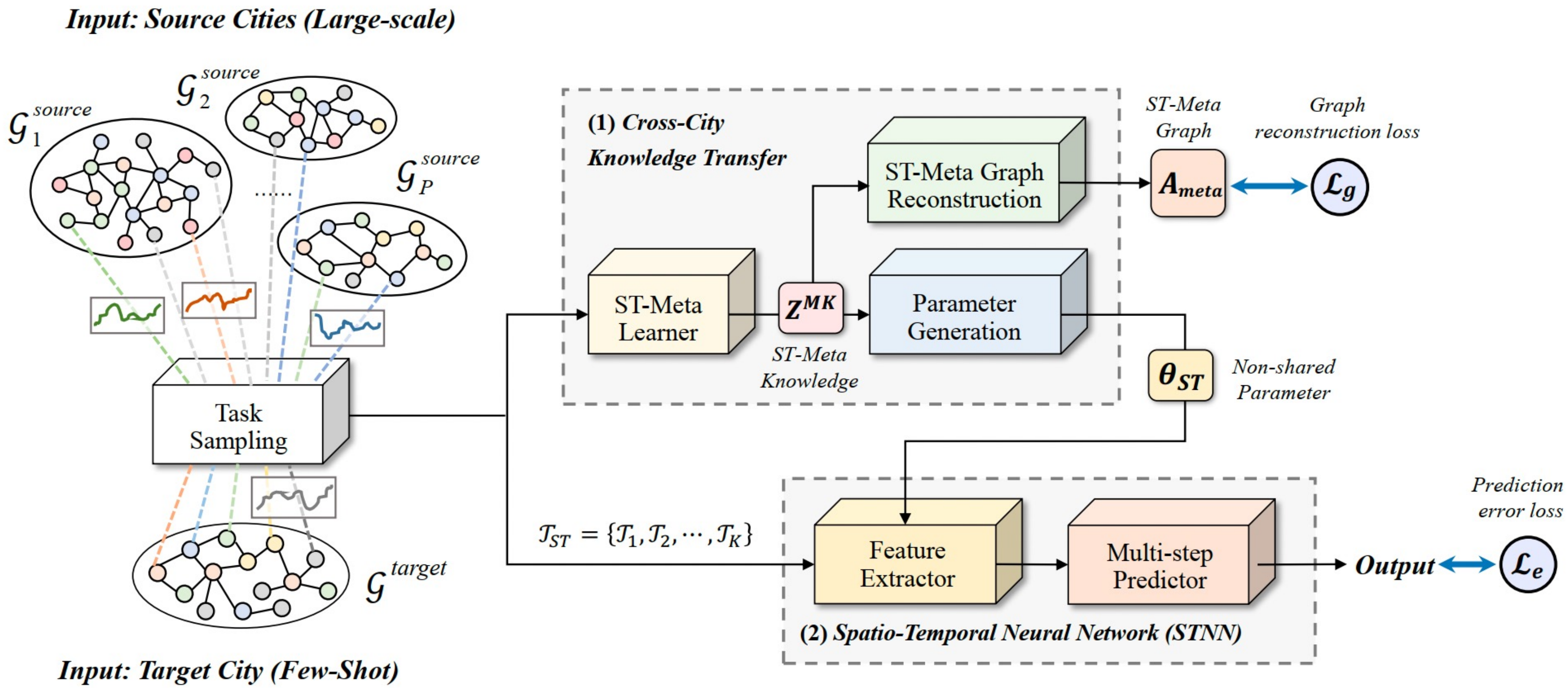
Motivation

- **Globally shared model → Non-shared model parameters**
  - Generate model parameters based on node-level meta knowledge
  - Node-level knowledge transfer through parameter matching (with similar spatio-temporal characteristics across source cities and target city)
- **Structure-aware learning**
  - Reconstruct the graph structure during training
  - Avoid structure deviation across multiple cities

Table of Contents

Methodology

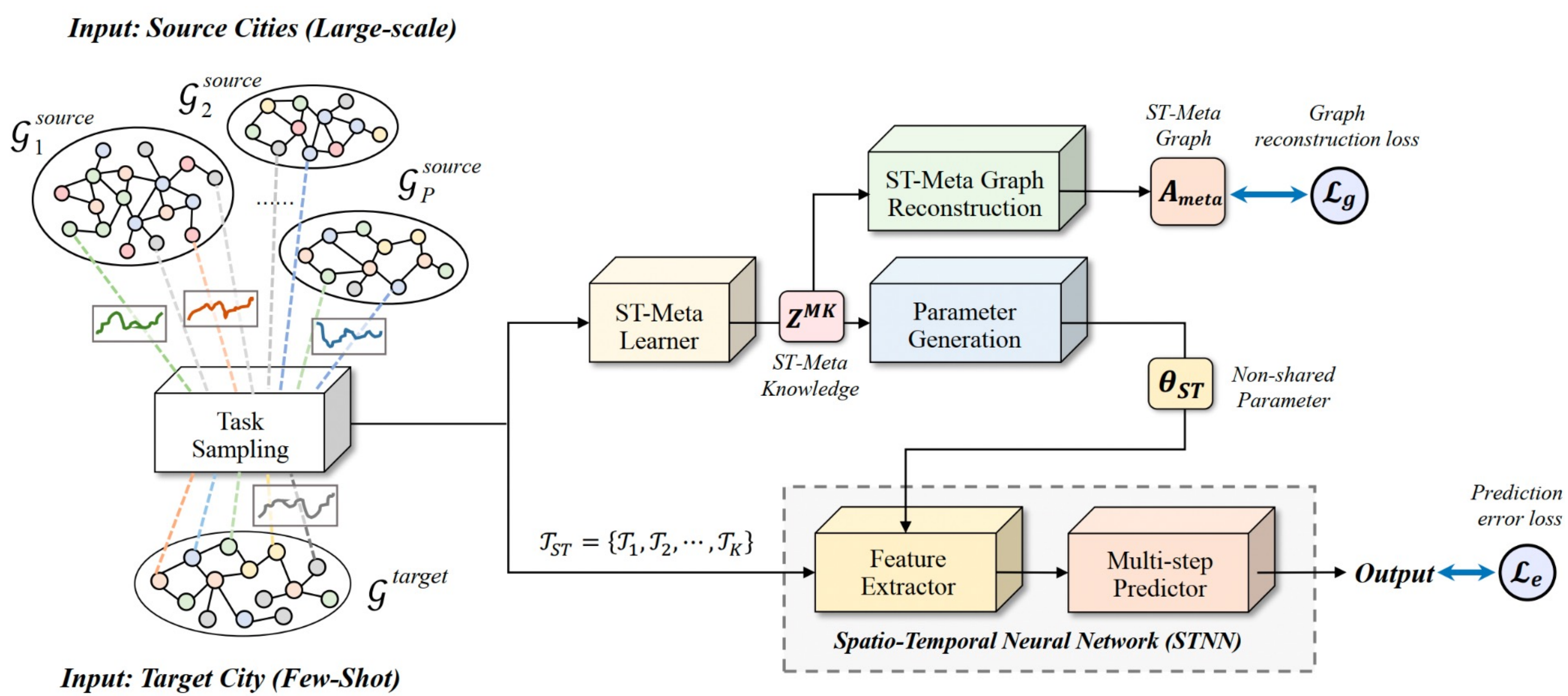
- **Overview of proposed ST-GFSL framework**



Spatio-Temporal Graph Few-Shot Learning (ST-GFSL)

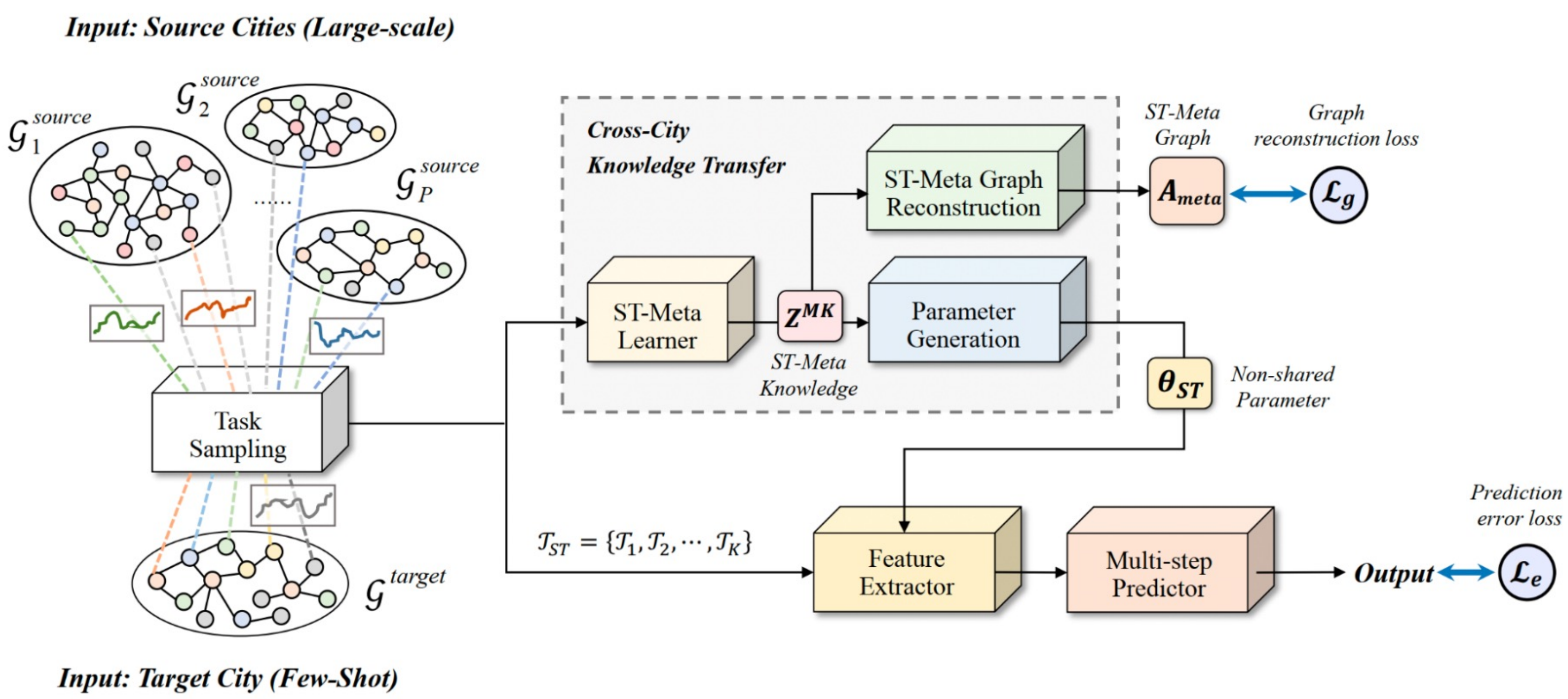
Methodology

- **Spatio-Temporal Neural Network (STNN)**
  - Backbone for feature extractor (time series model, graph neural network, hybrid model, etc.)



Methodology

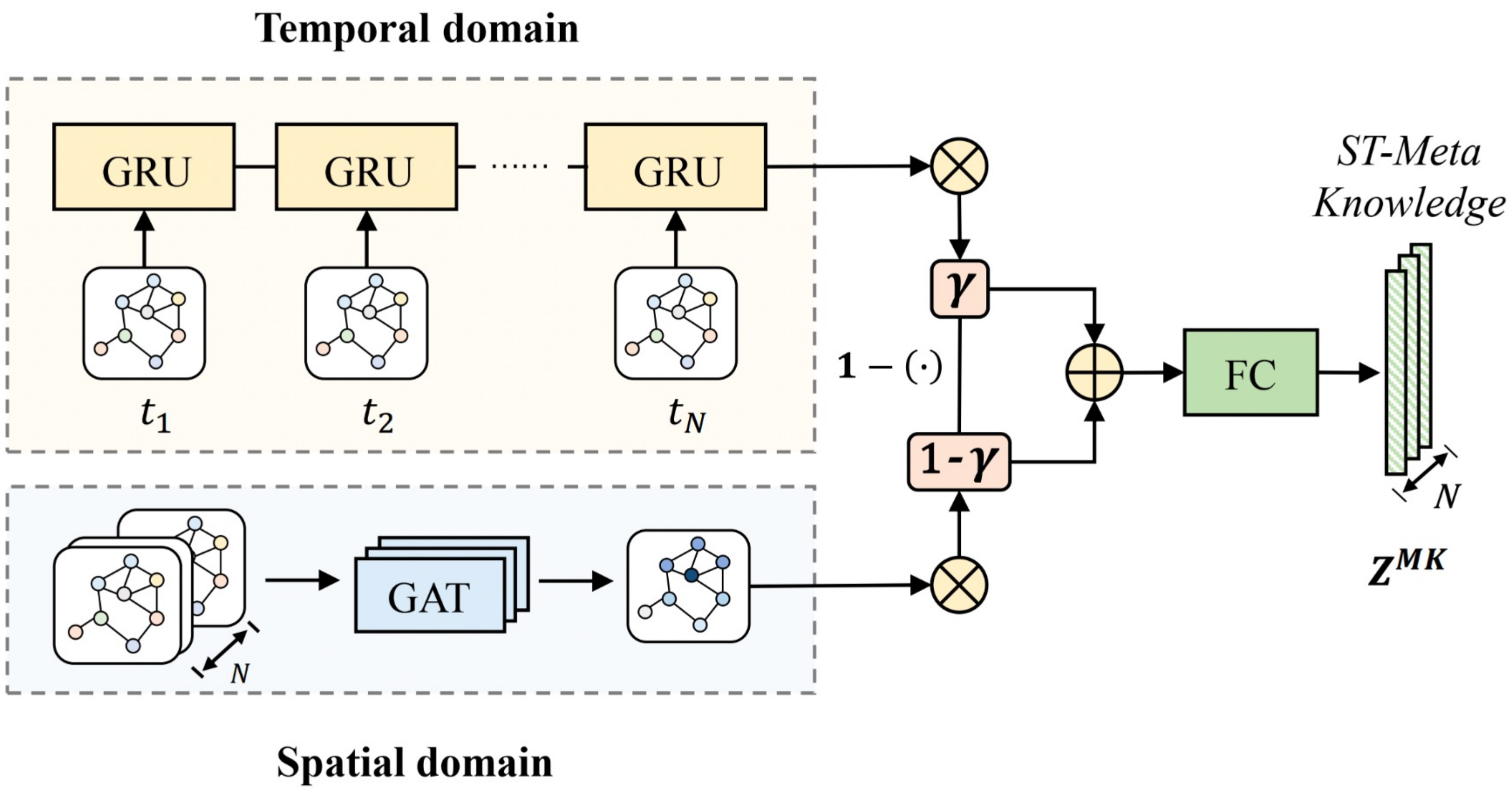
- **Cross-City Knowledge Transfer**
  - ST-Meta Learner: obtain the node-level meta knowledge
  - Parameter Generation: customize feature extraction among different cities
  - ST-Meta Graph Reconstruction: structure-aware learning



Methodology

- **ST-Meta Learner: Node-level meta knowledge**

$$\mathrm{Z}^{MK} = W^{\gamma}(\gamma \circ \mathrm{Z}^{tp} + (1-\gamma) \circ \mathrm{Z}^{sp})$$

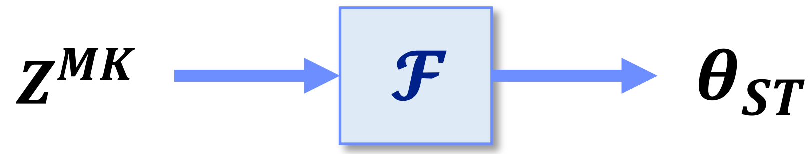


Methodology

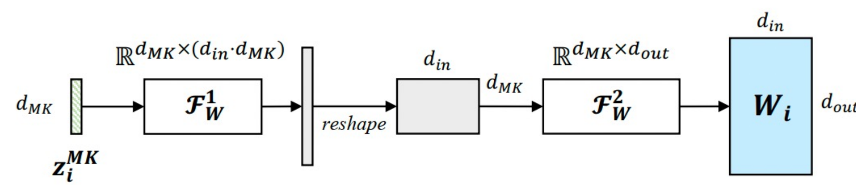
- **Parameter Generation**
  - Non-shared parameters of feature extraction
  - Similar meta knowledge feature → similar model parameter
  - Knowledge transfer via parameter matching

Methodology

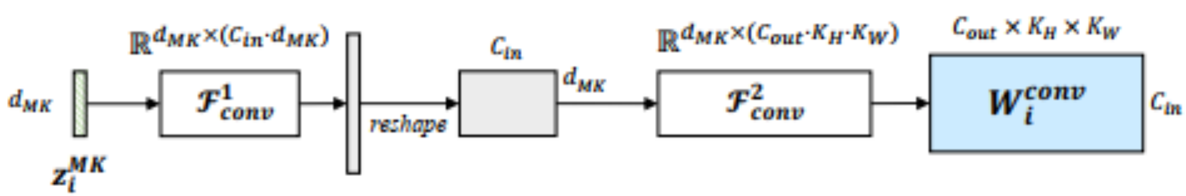
- **Parameter Generation**



  - Linear Layer $Y = \boldsymbol{W}X + \boldsymbol{b}$
    - $\boldsymbol{W} \in \mathbb{R}^{d_{out} \times d_{in}}$: two-step generation
      - $\mathcal{F}_1^W$: $\mathbb{R}^{d_{MK}} \rightarrow \mathbb{R}^{d_{in} \cdot d_{MK}} \rightarrow \mathbb{R}^{d_{in} \times d_{MK}}$
      - $\mathcal{F}_2^W$: $\mathbb{R}^{d_{in} \times d_{MK}} \rightarrow \mathbb{R}^{d_{in} \times d_{out}}$
    - $\boldsymbol{b} \in \mathbb{R}^{d_{out}}$
      - $\mathcal{F}_1^b$: $\mathbb{R}^{d_{MK}} \rightarrow \mathbb{R}^{d_{out}}$



  - Convolutional Layer
    - Two-step generation, details shown in our paper



Methodology

- **ST-Meta Graph Reconstruction**
  - Express the structural information of different graphs
  - Reduce structure deviation caused by different source data distribution
  - Predict the likelihood of an edge between v_i and v_j

$$p(a_{ij}|z_i^{MK}, z_j^{MK}) = sigmoid((z_i^{MK})^T, z_j^{MK})$$

  - Construct ST-Meta graph

$$\mathrm{A}_{meta} = sigmoid[(\mathrm{Z}^{\mathbf{MK}})^T \cdot \mathrm{Z}^{\mathbf{MK}}]$$

  - Structure-aware learning

$$\mathcal{L}_g = ||\mathrm{A}_{meta} - \mathrm{A}||^2$$

Methodology

- Learning process of ST-GFSL

Algorithm 1: ST-GFSL base-model meta training

Input: Source spatio-temporal Graph Dataset $\mathcal{G}_{ST}$,
learning rate hyperparameter α, β
Output: Trained ST-GFSL model parameters θ^*

```
θ ← random initalization;
// θ = θ_ST + θ_predictor
while not done do
    Sample a batch of tasks T_ST ← SampleTask(G_ST) ;
    for T_i ∈ T_ST do
        S_{T_i} ← K_S support set sample from T_i ;
        Q_{T_i} ← K_Q query set sample from T_i ;
        Evaluate ∇_θ L_{T_i}(f_θ) with S_{T_i} via Equation (11) ;
        Compute adapted parameters with gradient
          descent: θ'_{T_i} = θ − α∇_θ L_{T_i}(f_θ) ;
        Evaluate ∇_θ L_{T_i}(f(θ'_{T_i})) with Q_{T_i} via Equation (11)
          ;
    Update θ* ← θ − β∇_θ Σ_{T_i} L_{T_i}(f_{θ'_i})
```

Table of Contents

Experiment

- **Overview**
  - RQ1: How well does ST-GFSL perform against other baseline methods on different datasets?
  - RQ2: How well are different spatio-temporal prediction models adaptable under the ST-GFSL framework?
  - RQ3: How does each proposed component of model contribute to the performance of ST-GFSL?
  - RQ4: How does each major hyperparameter affect the performance of ST-GFSL?

Experiment

- **Dataset Description**
  - Z-Score normalization
  - Missing values are filled by the linear interpolation

	METR-LA	PEMS-Bay	DiDi-Chengdu	DiDi-Shenzhen
#Node	207	325	524	627
#Edge	1,722	2,694	1,120	4,845
Interval	5 min	5 min	10 min	10 min
Time Span	34,272	52,116	17,280	17,280
Mean	58.274	61.776	29.023	31.001
std	13.128	9.285	9.662	10.969

Experiment

● Model Comparison (RQ1)

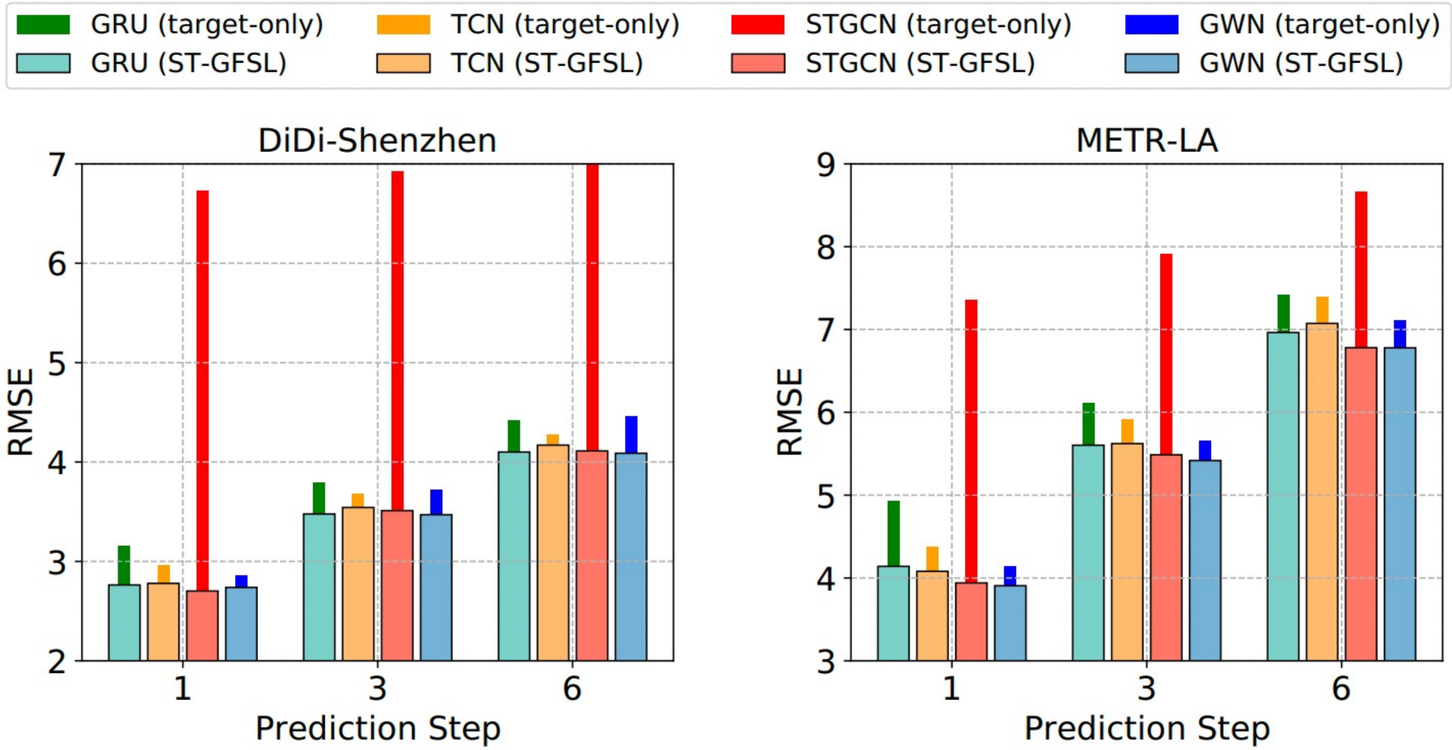
- Non-transfer Methods: Only use the few-shot data in target domain
- Transfer Methods: Transfer the knowledge from multiple source domains

Baselines	PEMS-BAY Dataset						METR-LA Dataset					
	MAE (↓)			RMSE (↓)			MAE (↓)			RMSE (↓)		
	5 min	15 min	30 min	5 min	15 min	30 min	5 min	15 min	30 min	5 min	15 min	30 min
HA	4.373	4.373	4.373	6.745	6.745	6.745	6.021	6.021	6.021	9.483	9.483	9.483
ARIMA	2.019	2.307	2.429	3.929	4.648	5.360	2.900	3.058	3.369	4.179	5.279	7.670
Target-only	1.556	1.920	2.368	3.092	4.043	5.153	2.740	3.229	3.860	4.924	6.118	7.417
Fine-tuned (Vanilla)	1.823	2.166	2.590	3.434	4.280	5.276	2.757	3.277	3.900	4.883	6.123	7.413
Fine-tuned (ST-Meta)	1.371	1.791	2.277	2.699	3.747	4.920	2.647	3.188	3.800	4.368	5.759	7.110
AdaRNN [29]	1.248	1.928	2.749	2.084	3.796	5.725	2.513	2.897	**3.312**	4.298	**5.567**	**6.732**
MAML [14]	1.081	1.600	2.141	1.906	3.291	4.708	2.405	2.960	3.639	4.159	5.710	7.124
ST-GFSL (ours)	**1.073**	**1.560**	**2.073**	**1.865**	**3.180**	**4.584**	**2.355**	**2.896**	3.557	**4.099**	5.588	6.961

Baselines	Didi-Chengdu Dataset						Didi-Shenzhen Dataset					
	MAE (↓)			RMSE (↓)			MAE (↓)			RMSE (↓)		
	10 min	30 min	60 min	10 min	30 min	60 min	10 min	30 min	60 min	10 min	30 min	60 min
HA	3.438	3.438	3.438	4.879	4.879	4.879	2.955	2.955	2.955	4.342	4.342	4.342
ARIMA	2.825	3.305	4.317	3.889	4.253	5.597	2.888	3.056	3.596	4.489	4.764	5.575
Target-only	2.386	2.700	3.085	3.516	4.017	4.569	2.071	2.454	2.834	3.154	3.793	4.422
Fine-tuned (Vanilla)	2.586	2.877	3.246	3.746	4.213	4.751	2.117	2.490	2.867	3.196	3.831	4.442
Fine-tuned (ST-Meta)	2.240	2.693	3.083	3.249	3.956	4.519	2.033	2.454	2.850	2.989	3.719	4.385
AdaRNN [29]	2.260	2.724	3.036	3.231	3.942	**4.324**	2.107	2.473	2.807	3.041	3.674	4.231
MAML [14]	2.215	2.599	2.956	3.215	3.858	4.399	1.917	2.330	2.673	2.825	3.546	4.158
ST-GFSL (ours)	**2.188**	**2.579**	**2.927**	**3.190**	**3.820**	4.339	**1.890**	**2.288**	**2.644**	**2.763**	**3.477**	**4.100**

Experiment

- **Performance Comparison of different feature extractor (RQ2)**
  - Use different feature extractor, including classical time series model and superior spatio-temporal graph neural network models



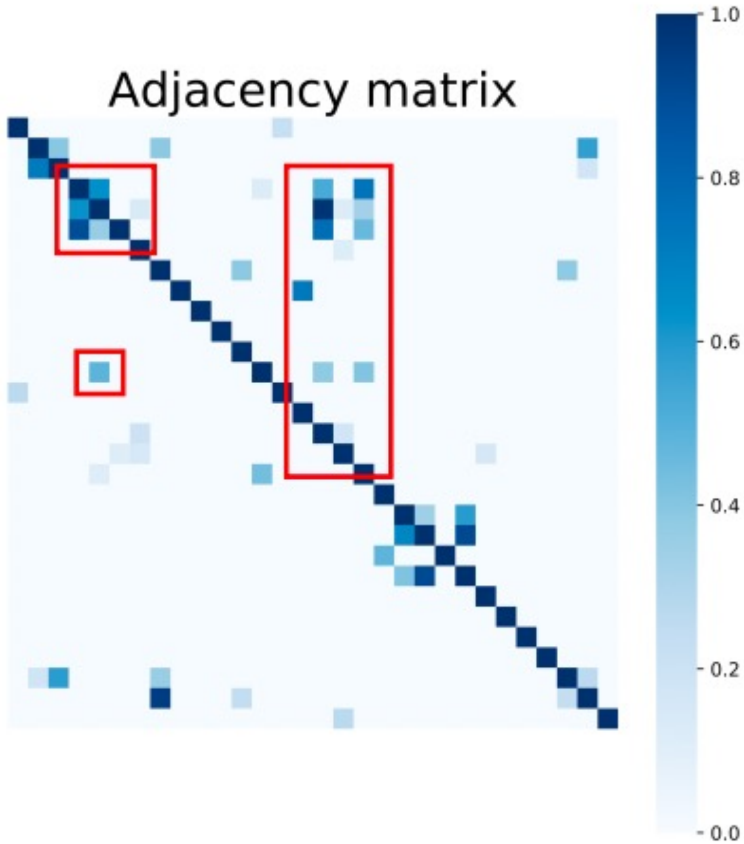
Experiment

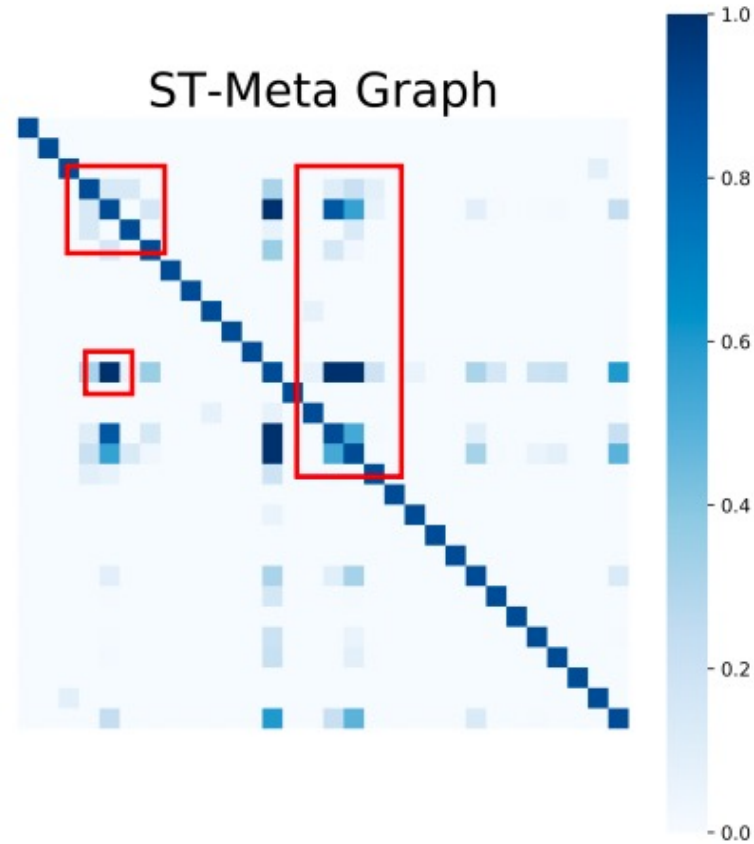
- **Ablation Study (RQ3)**

Ablation Methods	DiDi-Shenzhen		METR-LA	
	MAE (↓)		MAE (↓)	
	10 min	60 min	10 min	60 min
(M1a): Use temporal meta knowledge only	1.910	2.668	**2.332**	3.616
(M1b): Use spatial meta knowledge only	1.872	2.649	2.364	3.604
(M1c): Use random initialized vectors	1.937	2.680	2.422	3.697
(M2): Remove parameter generator	1.917	2.673	2.405	3.639
(M3): Remove graph reconstruction loss	2.286	3.000	3.087	4.140
ST-GFSL (Ours)	**1.856**	**2.634**	2.387	**3.546**

Experiment

- **Case study: Graph Reconstruction Loss (RQ3)**
  - Visual analysis on the original adjacency matrix and reconstructed ST-Meta graph





Experiment

- **Hyperparameter Study (RQ4)**

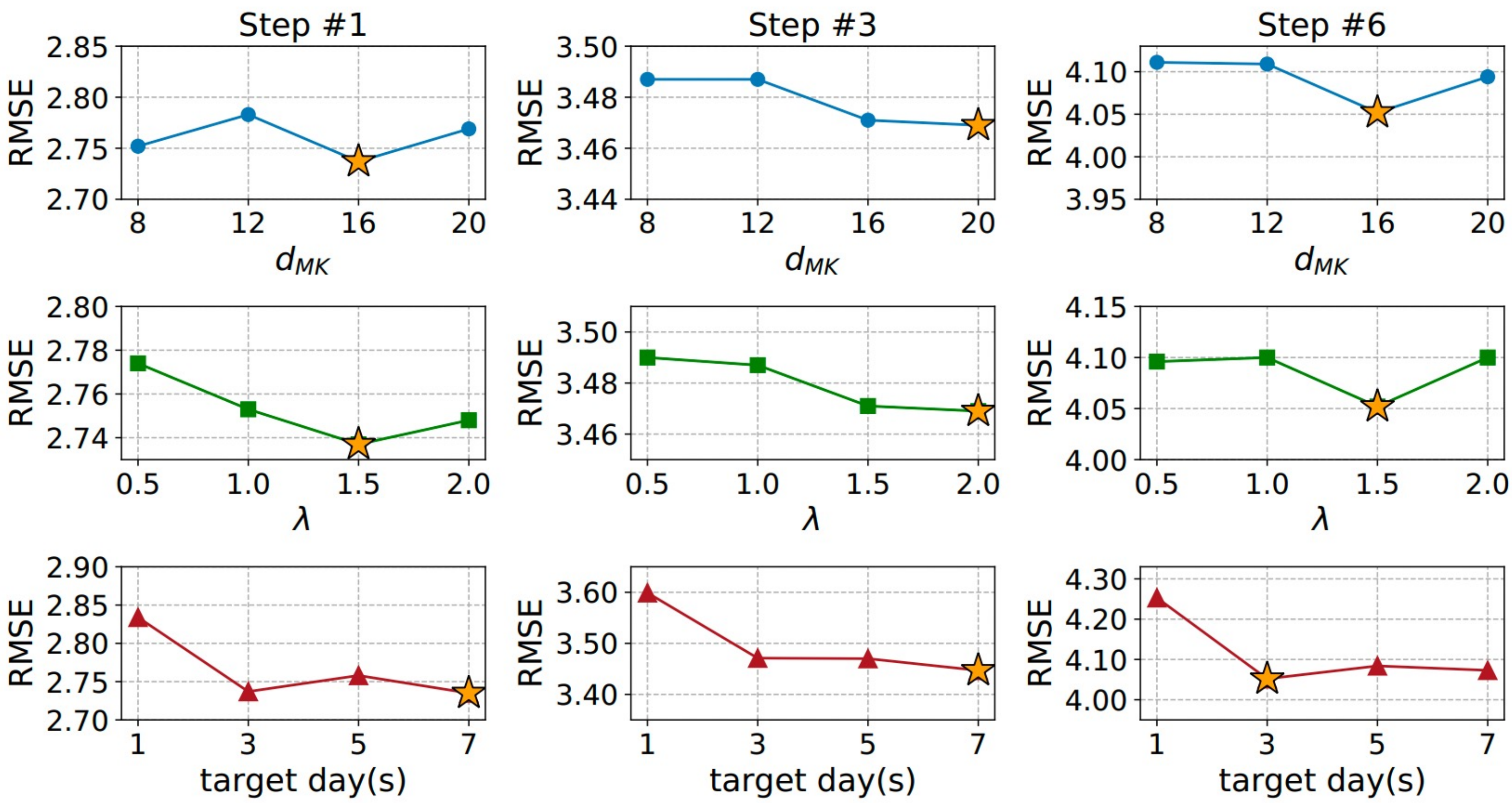


Table of Contents

Conclusion

- **Summary**
  - We firstly propose a spatio-temporal graph few-shot learning framework called ST-GFSL for cross-city knowledge transfer.
  - Extensive experimental results in the running case of traffic speed prediction demonstrate the superiority of ST-GFSL over other baseline methods.
    - parameter matching via meta knowledge transfer
    - structural-aware learning
- **Future Work**
  - Application on more spatio-temporal graph learning tasks
  - Explore the source domains selection problem

Thanks

Bin Lu
robinlu1209@sjtu.edu.cn