Network-Wide Traffic States Imputation Using Self-interested Coalitional Learning

Huiling Qin **Xianyuan Zhan** **Yuanxun Li** **Xiaodu Yang** **Yu Zheng**

Xidian University, Xi’an, China

JD Intelligent Cities Research, Beijing, China

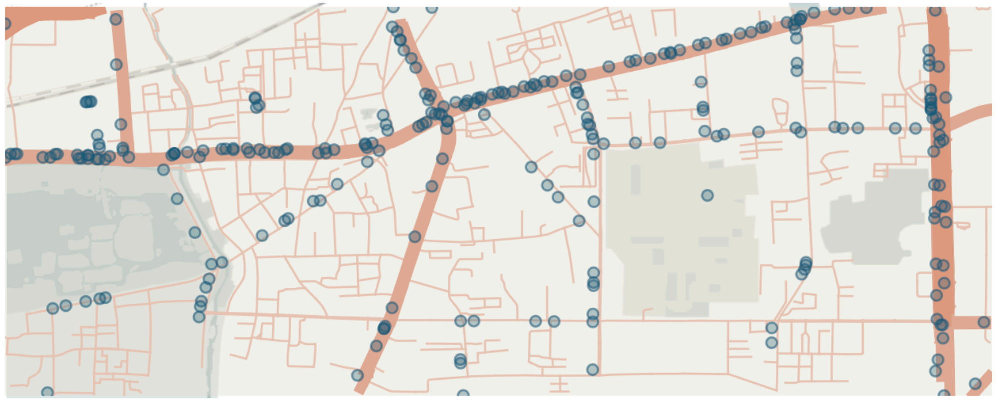
Sun Yat-sen University, Guangzhou, China

Southwest Jiaotong University, Chengdu, China

orekinana@gmail.com

*August 14th-18th, **KDD 2021 – Virtual Conference***

Motivation



(a) Vehicle trajectory points at 06:00 am in a small region of Jinan

(b) Vehicle trajectory points at 17:00 pm in a small region of Jinan

Figure 1: Spatio-temporal heterogeneity in trajectory data

What? →

Traveler information provision

Optimize timing configuration

Close certain roads on emergency events

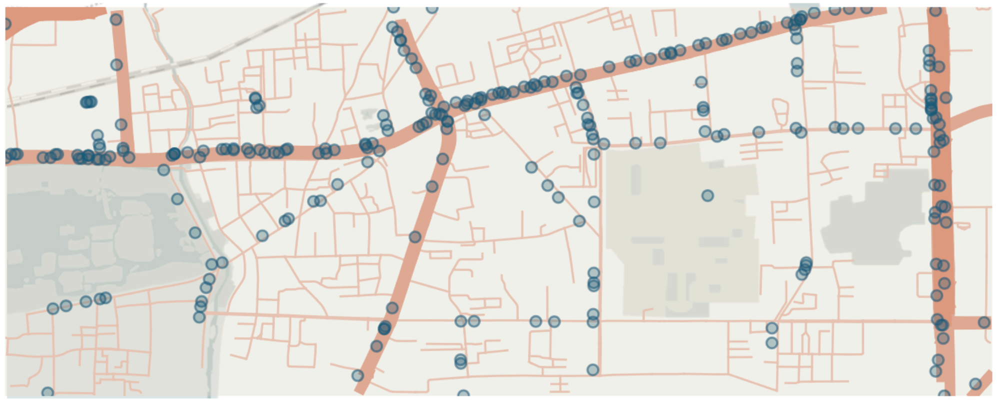
...

How? →

Reliable city-scale monitoring

Real-time inference

Challenges



(a) Vehicle trajectory points at 06:00 am in a small region of Jinan

(b) Vehicle trajectory points at 17:00 pm in a small region of Jinan

Figure 1: Spatio-temporal heterogeneity in trajectory data

For data:

- **Data sparsity**

[Taxis only account for a small faction of the total traffic]

- **Complex spatio-temporal pattern**

[Road network are influenced by the rhythm of the city]

- **Data unreliability**

[Traffic states are obtained from sample vehicle trajectories]

For model:

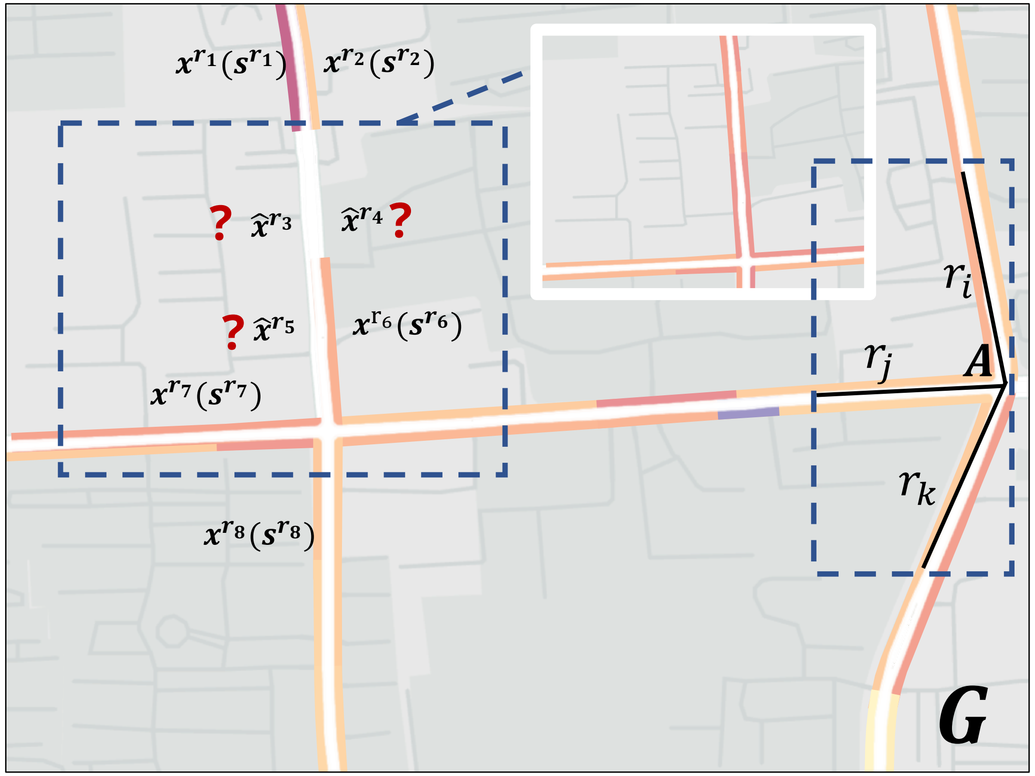
- **Interpretability**

[Given the reliability issue in the observed data, it is desired to have a model making a certain level of imputation interpretability]

- **Real-time inference**

[Real-world applications require update traffic information for the large road network as frequently as possible.]

Problem Definition



Definition:

Road adjacency graph: $G = \{R, A\}$

Traffic speed data: $X_{t_0:t_n} = [x^r_{t_0:t_n} | r \in R]$

Number of trajectories: $S_{t_0:t_n} = [s^r_{t_0:t_n} | r \in R]$

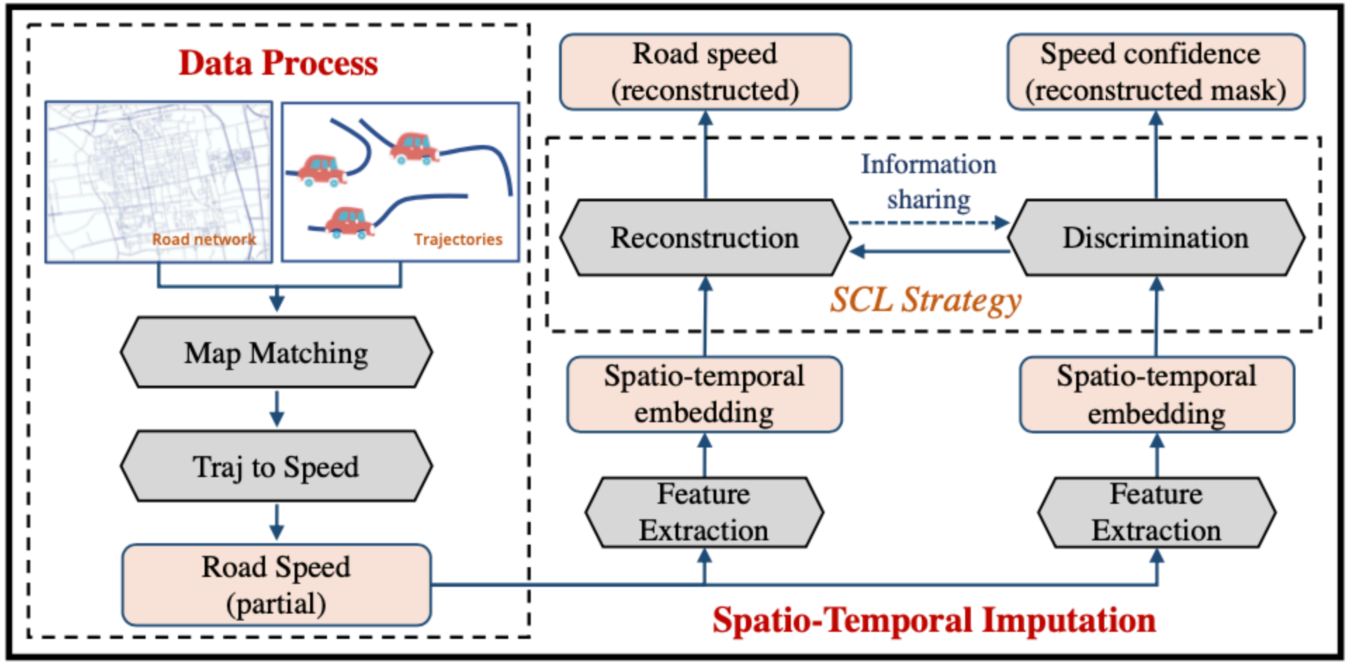
Observability mask: $M_{t_0:t_n} = [m^r_{t_0:t_n} | r \in R]$

Problem statement:

Impute all the missing entries in $X_{t_0:t_n}$ by constructing a filled matrix $\widehat{X}_{t_0:t_n}$, while providing the imputation confidence $P_{t_0:t_n} = [p^r_{t_0:t_n} | r \in R]$ of the results, which can be denoted as $\mathcal{F}(X, M) \mapsto [\widehat{X}, P]$.

Self-interested Coalitional Learning (SCL)

- **Overall framework:**



Main task: reconstructor f imputes complete $\hat{X}$ from observed X.

$$A : f(X) = \hat{X}, \qquad \mathcal{L}_A = loss_A(X, \hat{X})$$

$$B : d(X, \hat{X}) = P, \qquad \mathcal{L}_B = loss_B(P, M)$$

Companion task: Quantify the uncertainty / confidence given partially observed X and the filled data $\hat{X}$.

Self-interested Coalitional Learning (SCL)

Main task: reconstructor f imputes complete $\hat{X}$ from observed X.

$$A: f(X) = \hat{X}, \qquad \mathcal{L}_A = loss_A(X, \hat{X})$$

$$B: d(X, \hat{X}) = P, \qquad \mathcal{L}_B = loss_B(P, M)$$

Companion task: Quantify the uncertainty / confidence given partially observed X and the filled data $\hat{X}$.

- **Conventional approaches:**
  - Multi-task learning: Exploit the shared information and underlying commonalities between the two tasks.

    $$\min_{f,d} \lambda \cdot loss_A + (1-\lambda) \cdot loss_B, \quad \lambda \in (0,1)$$

  - Drawbacks:
    - X Two tasks may have some contradictions in some settings
    - X Tuning hyper-parameter λ is tricky
  - Adversarial leaning: Make two tasks learn against each other, thus improves the performance of both tasks.

    $$\min_f \max_d M \odot \log d(X, f(X)) + (1-M) \odot \log(1 - d(X, f(X)))$$

  - Drawbacks:
    - X $loss_a$ is not explicitly optimized, potential loss of information
    - X Notoriously hard to train

Self-interested Coalitional Learning (SCL)

Main task: reconstructor f imputes complete $\hat{X}$ from observed X.

$$A: f(X) = \hat{X}, \qquad \mathcal{L}_A = loss_A(X, \hat{X})$$

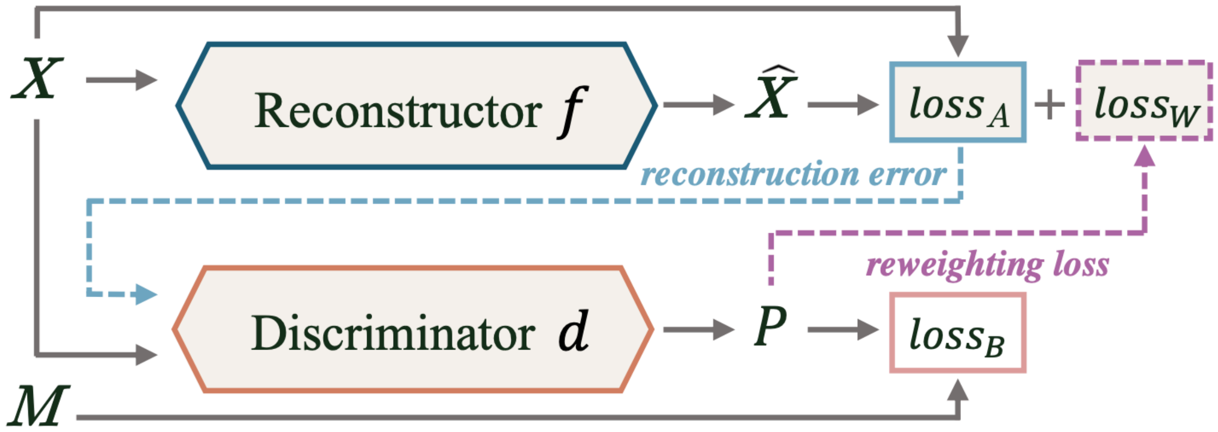
$$B: d(X, \hat{X}) = P, \qquad \mathcal{L}_B = loss_B(P, M)$$

Companion task: Quantify the uncertainty / confidence given partially observed X and the filled data $\hat{X}$.

- **An improved strategy: SCL**

$$A: f(X) = \hat{X}, \qquad \mathcal{L}_A = loss_A(X, \hat{X}) + loss_W(X, P)$$

$$B: d(X, g(X, \hat{X})) = P, \qquad \mathcal{L}_B = loss_B(P, M).$$



Self-interested Coalitional Learning (SCL)

$$\frac{\partial \mathcal{L}_B}{\partial d} = h_B = -\sum\left[M \oslash d\left(X, g\left(X, f(X)\right)\right) - (1-M) \oslash \left(1 - d\left(X, f(X)\right)\right)\right] \xrightarrow[\textit{impacts}]{\textit{Gradient}} \frac{\partial h_B}{\partial f}\ ?$$

$$\frac{\partial h_B}{\partial f} = \frac{\partial h_B}{\partial g} \cdot \frac{\partial g}{\partial f} = -\sum\left[M \oslash d\left(X, g\left(X, f(X)\right)\right) - (1-M) \oslash \left(1 - d\left(X, f(X)\right)\right)\right] \odot 2\left(X - f(X)\right) \odot \nabla f(X)$$

$$= \boxed{\sum_{x_i \in O} \frac{1}{p_i}(x_i - \widehat{x_i})^2 \nabla f(x_i) - \sum_{x_j \in U} \frac{1}{1-p_j}\left(x_j - \widehat{x_j}\right)^2 \nabla f(x_j)}$$

Gradient of a new loss term

$$loss_W(X, P) = \sum_{x_i \in O} \frac{1}{p_i}(x_i - \widehat{x_i})^2 - \sum_{x_j \in U} \frac{1}{1-p_j}\left(x_j - \widehat{x_j}\right)^2$$

Reweighting loss function

$$\mathcal{L}_A = \sum_{x_i \in O} \underbrace{\boxed{\left(1 + \frac{1}{p_i}\right)}(x_i - \widehat{x_i})^2 - \sum_{x_j \in U} \boxed{\frac{1}{1-p_j}}}_{\textit{Re-weighting factor}}\left(x_j - \widehat{x_j}\right)^2$$

Model Construction

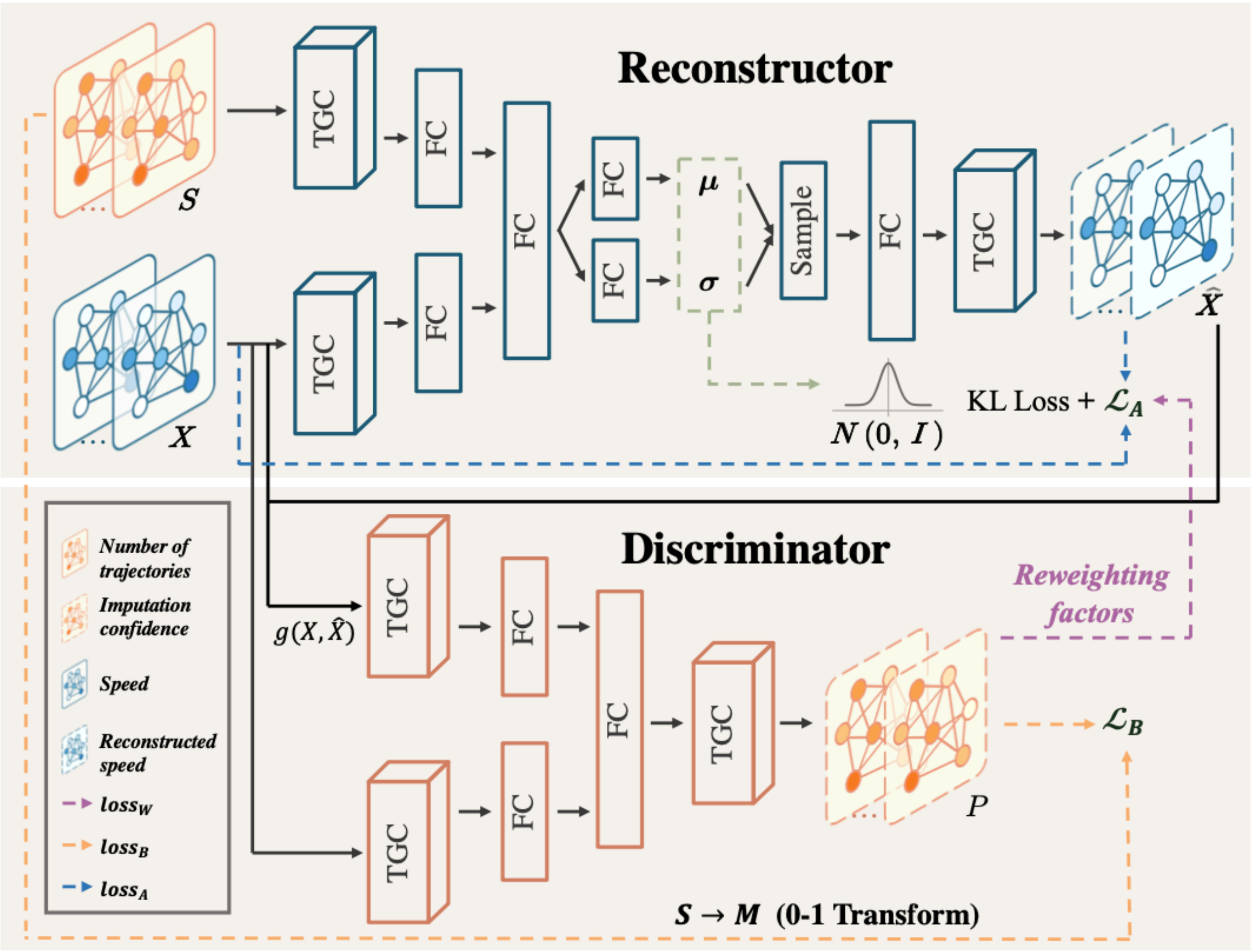


Figure 4: Detailed ST-SCL model design for our problem

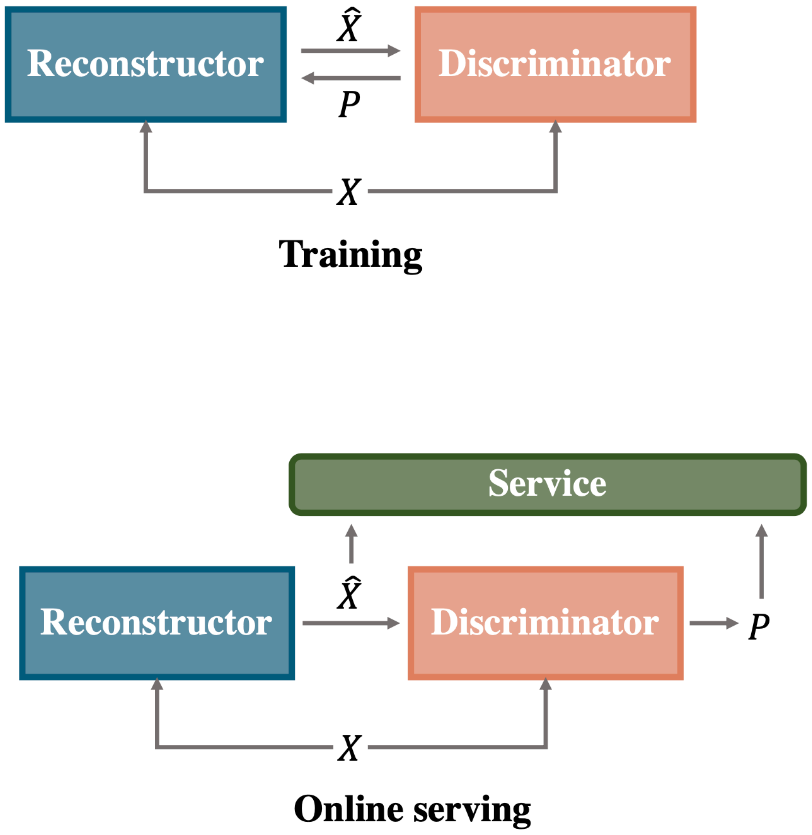


Figure 5: Training and online serving stages in ST-SCL

Experiments

Dataset

Road network: The simplify road network of Jinan comprises 2522 nodes and 608 edges.

Trajectories: GPS dataset over a period of 30 days. The average sampling rate is 3 seconds per point.

Settings: Time slot set as 5 minutes and partition the data into 28 days of training and 2 days for evaluation.

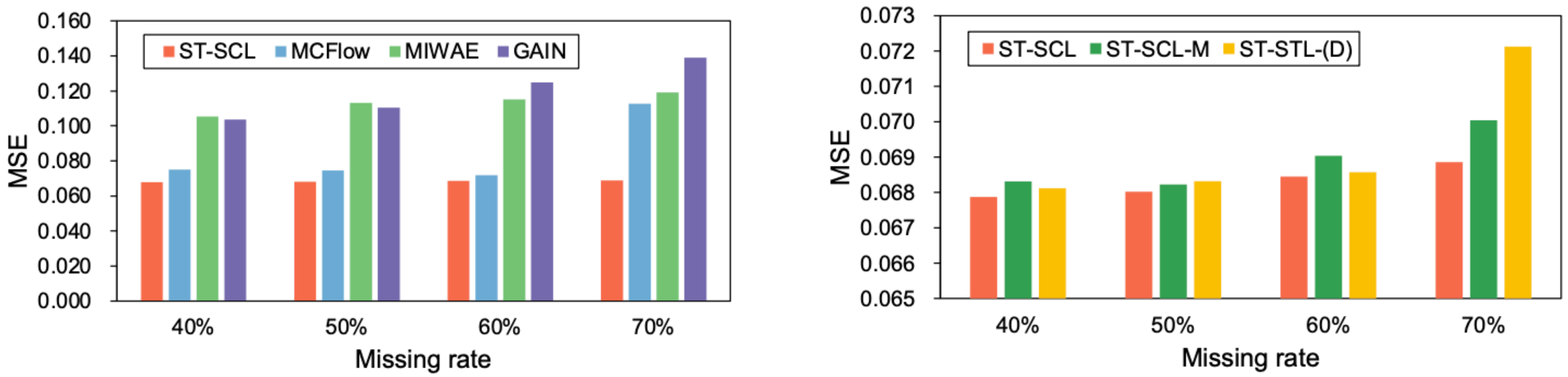
Time period	Time	Missing rate
All day	00:00-24:00	37.9%
Morning peak	06:00-10:00	15.4%
Evening peak	10:00-18:00	17.1%
Flat peak	18:00-20:00	27.0%
Night hour	20:00-06:00	77.2%

Experiments

Table 1: Evaluation results of ST-SCL and the baseline methods for morning and evening peak, flat peak, and night hours

Methods	Overall		Morning peak		Evening peak		Off peak		Night	
	MSE	RMSE	MSE	RMSE	MSE	RMSE	MSE	RMSE	MSE	RMSE
GAIN	0.1035	0.3217	0.0993	0.3151	0.0889	0.2982	0.0886	0.2977	0.2095	0.4577
MIWAE	0.1053	0.3245	0.1065	0.3263	0.1114	0.3338	0.1018	0.3191	0.0983	0.3135
MCFlow	0.0751	0.2740	0.0807	0.2841	0.0761	0.2759	0.0726	0.2694	0.0773	0.2780
MF	0.1314	0.3625	0.1326	0.3641	0.1186	0.3444	0.1182	0.3438	0.2295	0.4791
ST-SCL	**0.0679**	**0.2605**	**0.0740**	**0.2720**	**0.0697**	**0.2640**	**0.0677**	**0.2601**	**0.0617**	**0.2483**
ST-SCL-M	0.0703	0.2651	0.0752	0.2742	0.0718	0.2680	0.0683	0.2613	0.0725	0.2693
ST-SCL-G	0.1518	0.3896	0.1486	0.3854	0.1469	0.3832	0.1511	0.3887	0.1612	0.4014
ST-SCL(-D)	0.0683	0.2613	0.0747	0.2733	0.0704	0.2653	0.0678	0.2603	0.0622	0.2493
ST-SCL(-V)	0.0695	0.2636	0.0754	0.2746	0.0724	0.2691	0.0695	0.2636	0.0631	0.2512

Experiments

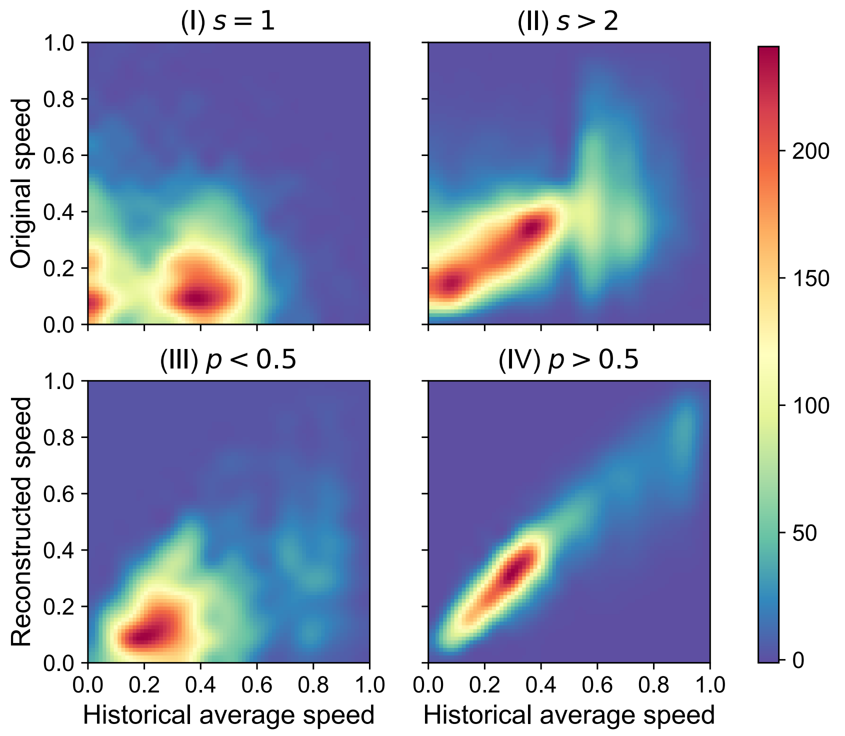


(a) The performance of ST-SCL and baselines under different missing rate

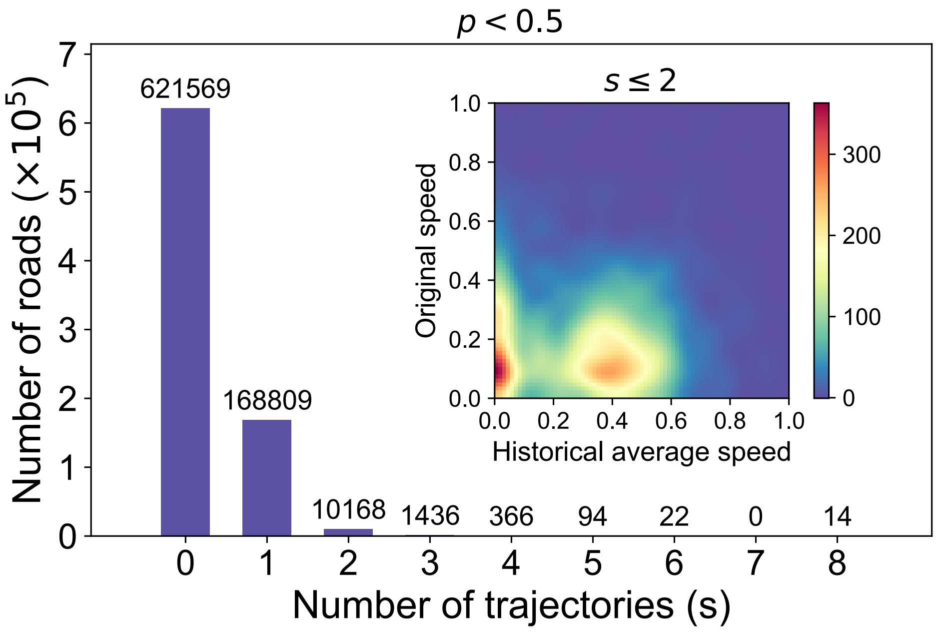
(b) The performance of ST-SCL and its variants under different missing rate

Figure 6: The performance under different missing rate

Experiments



(a) Heatmap of normalized historical average and original/reconstructed speeds



(b) Analysis on trajectory supports of road speeds with p < 0.5

Figure 7: Relationships of normalized historical average and original/reconstructed speeds under different trajectory supports and confidence levels.

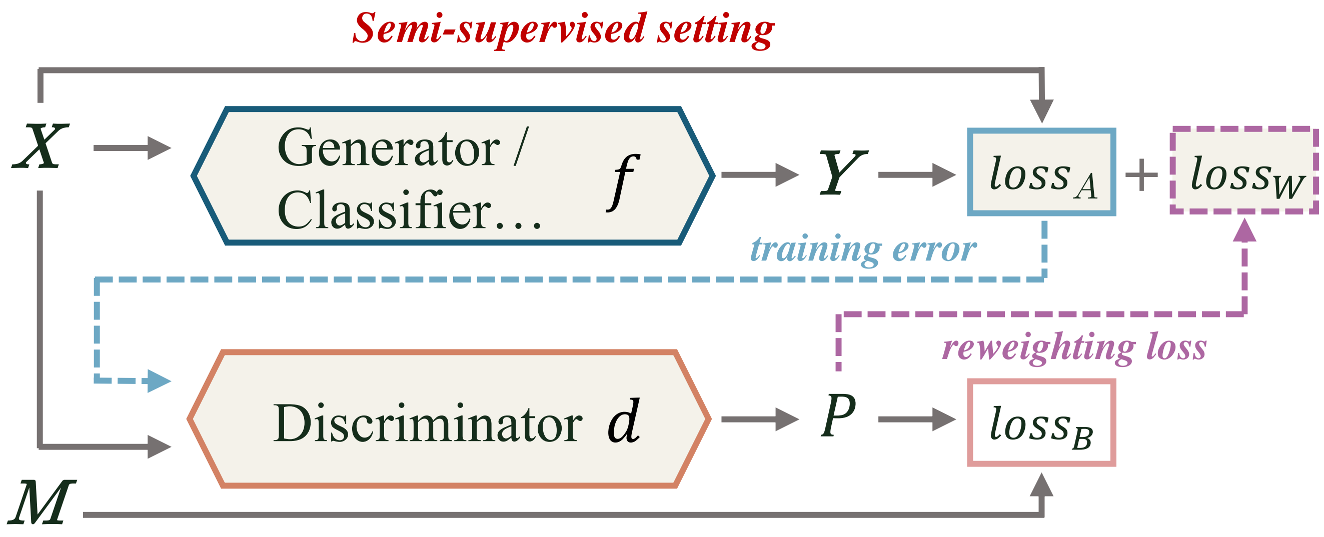
Experiments

(c) Estimated speed

Figure 8: Comparison among historical average, original, and estimated speed in Jinan at 2017/09/02 21:00

Conclusion and Perspective

- We propose the ST-SCL, a new framework that performs real-time network-wide traffic state imputation with partially observed data, while providing interpretable confidence on the results.
- We develop a novel self-interested coalitional learning (SCL) scheme that can boost the performance of a semi-supervised task by forge cooperation with an extra discriminator in a self-interested manner.
- We design highly customized reconstructor and discriminator for the traffic state imputation problem.



$$\mathcal{F}(\boldsymbol{X}, \boldsymbol{M}) \mapsto [\boldsymbol{Y}, \boldsymbol{P}]$$

$A: f(X) = Y$ $\quad\quad$ $\mathcal{L}_A = loss_A(X, Y) + loss_W(X, P)$

$B: d\big(X, g(X, Y)\big) = P$ $\quad\quad$ $\mathcal{L}_B = loss_B(P, M)$

Thanks!