Efficient Traffic Prediction Through Spatio-Temporal Distillation

Qianru Zhang [1], Xinyi Gao [2], Haixin Wang [3], Siu-Ming Yiu [1*], Hongzhi Yin [2†]

[1] The University of Hong Kong
[2] The University of Queensland
[3] University of California, Los Angles

Abstract

Graph neural networks (GNNs) have gained considerable attention in recent years for traffic flow prediction due to their ability to learn spatio-temporal pattern representations through a graph-based message-passing framework. Although GNNs have shown great promise in handling traffic datasets, their deployment in real-life applications has been hindered by scalability constraints arising from high-order message passing. Additionally, the over-smoothing problem of GNNs may lead to indistinguishable region representations as the number of layers increases, resulting in performance degradation. To address these challenges, we propose a new knowledge distillation paradigm termed LightST that transfers spatial and temporal knowledge from a high-capacity teacher to a lightweight student. Specifically, we introduce a spatio-temporal knowledge distillation framework that helps student MLPs capture graph-structured global spatio-temporal patterns while alleviating the over-smoothing effect with adaptive knowledge distillation. Extensive experiments verify that LightST significantly speeds up traffic flow predictions by 5X to 40X compared to state-of-the-art spatio-temporal GNNs, all while maintaining superior accuracy.

Introduction

Recent advancements in intelligent transportation systems have seen significant progress in traffic flow prediction (Wang et al. 2020; Pan et al. 2019; Liang et al. 2019) through the development of Graph Neural Networks (GNNs). GNN-based methods (Gao et al. 2024b,a,c) utilize the message-passing mechanism to propagate embeddings, enabling them to capture spatio-temporal traffic patterns. For example, STGCN (Yu, Yin, and Zhu 2018), AGCRN (Bai, Yao et al. 2020) and GWN (Shleifer, McCreery, and Chitters 2019) are built upon GNNs for traffic prediction (Zhang et al. 2024). GCN-based models (Yu, Yin, and Zhu 2018; Shleifer, McCreery, and Chitters 2019) use convolutional operations on the graph structure to extract spatial features of traffic data, while GAT-based methods (Han and Gong 2022) employ attention mechanisms to weigh the importance of graph-based neighboring locations for each region, propagating information.

The effectiveness of spatio-temporal GNNs is largely attributed to the complex model structure and recursive message-passing architecture that encodes high-order region-wise connectives and learns region representations. However, the increasing complexity of larger and deeper GNN model structures leads to the computationally intensive inference procedure, posing challenges for practical applications due to the scalability constraints. Therefore, a lightweight yet effective traffic prediction framework is required for practical settings of intelligent transportation systems. Additionally, propagating embeddings across multiple layers gradually makes node features more uniform, ultimately reducing the model's ability to differentiate node features (Chen et al. 2020; Zhou et al. 2020). This inherent recursive message-passing paradigm may fall short in encoding diverse spatio-temporal patterns, ultimately degrading traffic prediction performance.

To mitigate these challenges, existing methods (Izadi, Safayani, and Mirzaei 2024; Wang et al. 2024; Zhang et al. 2022) leverage knowledge distillation (KD) to transfer knowledge from complex teacher models to smaller student models, helping to overcome limitations in spatio-temporal graph neural networks (ST-GNNs). However, they still face challenges and achieve sub-optimal performances. KD-pruning method (Izadi, Safayani, and Mirzaei 2024) calculates pruning scores via cost function and fine-tunes the student network decomposed with GNNs, but they do not fully address the over-smoothing issue inherent to GNNs. Furthermore, incorporating GNNs into the student network does not effectively resolve the high computational cost associated with training and deploying these models. A recent work (Wang et al. 2024) proposes a Spatial-Temporal Knowledge Distillation (STKD) algorithm framework for lightweight network traffic anomaly detection, integrating multi-scale 1D CNNs and LSTMs with identity mapping for performance enhancement. However, the individual components of STKD, namely 1D CNNs and LSTMs, may not be optimal for capturing temporal correlations, potentially limiting its effectiveness in spatio-temporal domains. Firstly, 1D CNNs function within fixed-size windows, potentially impeding their ability to capture essential long-range dependencies critical for modeling intricate temporal patterns within traffic data. Secondly, both 1D CNNs and LSTMs generate fixed-length representations, potentially compro-

[*] Corresponding Author

[†] Corresponding Author

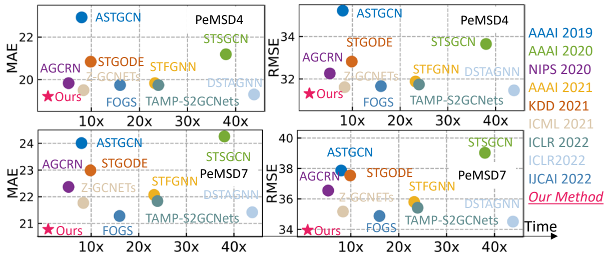

Figure 1: Model performance comparison in terms of both traffic flow prediction accuracy and inference time. Lower MAE and RMSE indicate better performance. The symbol 40× indicates that our LightST runs 40 times faster than a reference baseline method measured by inference time.

mising detailed temporal nuances essential for capturing subtle variations and complex temporal relationships in dynamic traffic environments.

In light of the motivations described above, this study aims to: i) develop a traffic prediction model that is highly scalable while effectively capturing complex spatio-temporal dependency patterns across various locations and time periods; and ii) enhance the spatio-temporal encoding function to mitigate the issue of over-smoothing. To achieve this, we propose a dual-level spatio-temporal knowledge distillation paradigm that effectively transfers complex dynamic spatio-temporal dependency knowledge into a compact and fast-to-execute student model. Specifically, in the distillation procedure, the soft prediction labels from the teacher GNN guide the learning of the student model. This transfer of knowledge effectively incorporates structural information from both spatial and temporal aspects. Furthermore, to avoid using a uniform alignment factor for all region pairs, we propose an adaptive embedding-level distillation framework that enhances the knowledge transfer while mitigating over-smoothing effects. Our empirical studies, as shown in Figure 1, suggest that our designed spatio-temporal knowledge distillation substantially benefits traffic prediction performance in terms of both forecasting accuracy and efficiency. Our key contributions are listed as follows:

- To overcome the computational and oversmoothing challenges in state-of-the-art GNN-based traffic prediction models, we propose distilling the complex spatio-temporal GNN architecture into a streamlined MLP model, enhancing both efficiency and robustness in traffic prediction.
- We design a new spatio-temporal knowledge distillation paradigm with two model alignment levels, enabling the transfer of knowledge related to spatial and temporal dynamics while enhancing the student model with a global context. Additionally, we propose an adaptive contrastive distillation scheme to further enhance the robustness of the prediction model against the over-smoothing issue.
- We empirically validate our new framework on 5 real-world traffic datasets. Evaluation results demonstrate that our model achieves state-of-the-art traffic prediction accuracy, with a 5× to 40× inference speedup compared to existing baselines. Our codes are available at: https://github.com/lizzyhku/TP/tree/main. We also attached the reproducibility checklist.

Methodology

Spatio-temporal Graph and Traffic Data

Following established practices (Lan et al. 2022; Chen, Segovia, and Gel 2021), we define our spatial graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the geographically-adjacent relationships between traffic sensing regions. We represent the traffic volume data across both spatial and temporal dimensions using a matrix $\mathbf{X} \in \mathbb{R}^{N \times T}$, where N denotes the number of regions and T represents the number of time slots. Each element $x_{n,t}$ within $\mathbf{X}$ corresponds to the traffic volume information of region n during time slot t.

Problem Formulation. Our goal is to develop a function $\mathcal{F}$ that predicts future traffic flow $\hat{\mathbf{Y}} \in \mathbb{R}^{N \times H}$ based on observed traffic flow data $\mathbf{X} = (x_1, ..., x_T) \in \mathbb{R}^{N \times T}$. The ground truth is $\mathbf{Y} = (x_{T+1}, ..., x_{T+H}) \in \mathbb{R}^{N \times H}$. Here, $\mathbf{X}$ represents the observed traffic flow from N sensing regions within a sensor graph $\mathcal{G}$ over the preceding T time slots, and $\hat{\mathbf{Y}}$ denotes the predicted traffic flow for the subsequent H time steps. Our approach aims to learn $\mathcal{F}$ while preserving both spatial and temporal dependencies within the data. This is shown as $\hat{\mathbf{Y}} = \mathcal{F}(\mathbf{X}; \mathcal{G})$.

Spatio-Temporal Graph Neural Networks

Data Scale. Graph Neural Networks (GNNs) have demonstrated significant potential in learning from spatio-temporal data (Li et al. 2018; Lan et al. 2022; Rao et al. 2022). Inspired by this, our teacher model leverages a graph-based message passing framework that harnesses the power of GNNs to capture region-wise dependencies. We begin by mapping traffic data into a latent representation space. Each element $\mathbf{X}_{n,t} \in \mathbf{X}$ is encoded into an embedding $\mathbf{E}^{(\mathrm{d})}_{n,t} \in \mathbb{R}^d$ as follows:

$$\mathbf{E}^{(\mathrm{d})}_{n,t} = \text{Z-Score}(x_{n,t}) \cdot \mathbf{e} = \frac{x_{n,t} - \mu}{\sigma} \cdot \mathbf{e}. \tag{1}$$

We normalize the value of $\mathbf{X}_{n,t}$ using the Z-Score function, which centers the data around zero with a standard deviation of one. This normalized value is then multiplied by a base embedding vector $\mathbf{e} \in \mathbb{R}^d$, resulting in the final embedding $\mathbf{E}^{(\mathrm{d})}_{n,t} \in \mathbb{R}^d$. μ represents the average traffic flow value of node n over the last 12 time steps. And σ denotes the standard deviation of the traffic flow values of node n over the previous 12 time steps. The base embedding vector $\mathbf{e}$ acts as a template, and the Z-Score normalization scales the traffic flow value, effectively creating a unique embedding for each data point based on its relative position within the normalized distribution.

Time-aware Spatial Message Passing. To capture time-aware spatial dependencies and learn representations specific to each time slot t, we employ a time-specific message passing mechanism among traffic sensing regions. This process involves aggregating information from neighboring

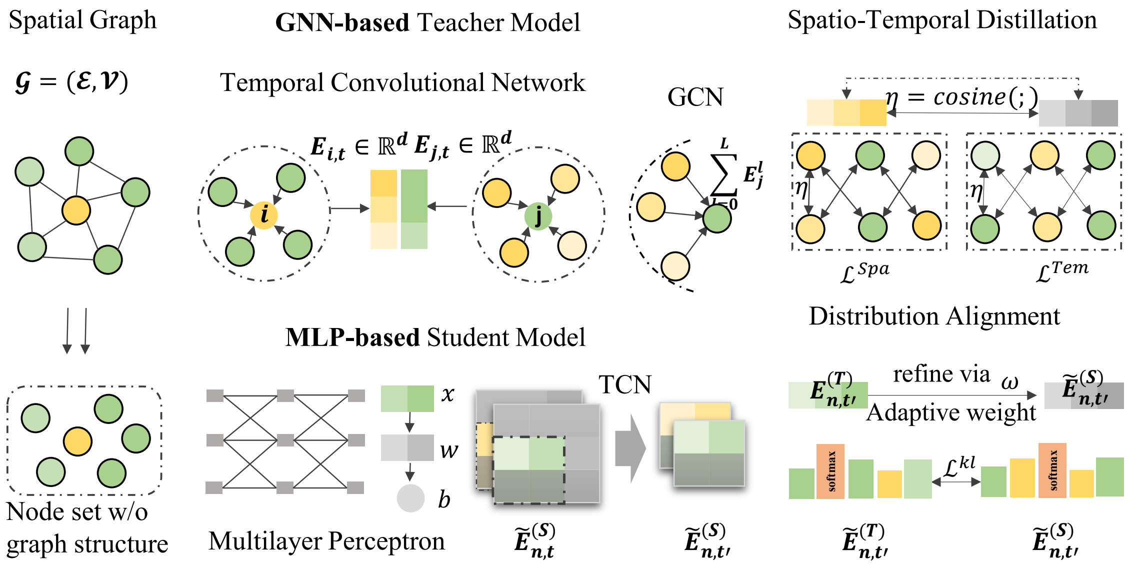

Figure 2: Our proposed spatio-temporal knowledge distillation framework, LightST, comprises two main components: a GNN-based teacher model and an MLP-based student model. The distillation paradigm itself is structured in two parts: spatio-temporal distillation and distribution alignment.

nodes within the sensor graph. The embedding of region n at the l-th GNN layer is denoted by $\mathbf{E}_n^{(l)}$, and is shown as follows:

$$\mathbf{E}_n^{(l)} = \sigma^{(1)}\Big(\sum_{j \in \mathcal{N}_n} \alpha_{n,j} \mathbf{W}^{(l-1)} \mathbf{E}_j^{(l-1)}\Big), \qquad (2)$$
$$\alpha_{n,j} = \frac{1}{\sqrt{|\mathcal{N}_n||\mathcal{N}_j|}}.$$

Here, $\mathcal{N}_n$ and $\mathcal{N}_j$ represent the sets of neighboring nodes for regions n and j, respectively. The embedding propagation mechanism aggregates information from these neighboring nodes. $\sigma^{(1)}(\cdot)$ represents the ReLU activation function. The normalization weight $\alpha_{n,j} \in \mathbb{R}$ for a node pair (n, j) is calculated based on the degrees of the nodes. This cross-layer information propagation and aggregation can be formalized in matrix form using the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ of our spatial graph $\mathcal{G}$:

$$\mathbf{E}^{(l)} = \sigma^{(1)}(\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{E}^{(l-1)} \mathbf{W}^{(l-1)\top}). \qquad (3)$$

The embedding matrix $\mathbf{E} = \sum_{l=0}^{L} \mathbf{E}^{(l)} \in \mathbb{R}^{N \times d}$ contains node embeddings, where each row represents individual traffic sensing point $\mathbf{E}_n$ $(1 \leq n \leq N)$. $\mathbf{D} \in \mathbb{R}^{N \times N}$ denotes the diagonal degree matrix, and L represents the number of graph neural iterations. The final aggregated region representations are given by $\mathbf{E} \in \mathbb{R}^{N \times d}$.

Temporal Encoder Layer. To capture temporal dependencies across all sensor regions, we employ a two-layer Temporal Convolutional Network (TCN) to model temporal correlations. This can be represented as follows:

$$\tilde{\mathbf{E}}_n = \sigma^{(2)}(\delta(\mathbf{W} * \mathbf{E}_n + \mathbf{b}) + \mathbf{E}_t), t \in [1, T]. \qquad (4)$$

Here, the embedding matrix of all N regions at the time slot of t is represented by $\mathbf{E}_t \in \mathbb{R}^{N \times d}$. For a specific region n, the embedding matrix is represented by $\mathbf{E}_n \in \mathbb{R}^{T \times d}$, which captures the embedding of region n across the previous T time slots. The temporal convolution kernel and bias are represented by $\mathbf{W} \in \mathbb{R}^{f \times d}$ and $\mathbf{b} \in \mathbb{R}^d$, respectively. These are learnable transformation parameters. Here, f denotes the kernel size. The operation $*$ denotes the temporal convolution, while the dropout $\delta(\cdot)$ and LeakyReLU activation function $\sigma^{(2)}(\cdot)$ are used as well. The output representation $\tilde{\mathbf{E}} \in \mathbb{R}^{N \times T \times d}$ is generated with two-layer TCNs that serve as the temporal encoder in the teacher model, so as to capture the time-evolving traffic patterns.

Distillation Process

This research aims to develop a *lightweight* and *efficient* traffic predictor that leverages spatial and temporal knowledge extracted from a Graph Neural Network (GNN) teacher model to achieve accurate traffic predictions. This is achieved through both explicit spatio-temporal distillation and implicit knowledge transfer from the GNN teacher model to an MLP-based student model via distribution alignment. We provide details of the distillation process as follows:

Spatio-Temporal Distillation. During this stage, we aim to align the probability distributions of the teacher and student models, thereby enhancing the predictive performance of the student model. This is achieved through knowledge distillation, where the output results of the teacher model are used as soft labels to guide the student model. We minimize the difference between the probability distributions of the teacher and student models by employing an MSE-based loss function (Wu et al. 2020). The MSE-based loss functions for the teacher and student models in traffic prediction are defined as follows:

$$\begin{aligned}\mathcal{L}^{(\mathrm{T})} &= \frac{1}{N}\sum_{n=1}^{N}\sum_{t'=T+1}^{T+H}(\hat{\mathbf{Y}}_{n,t'}^{(\mathrm{T})} - \mathbf{Y}_{n,t'})^2,\\ \mathcal{L}^{(\mathrm{S})} &= \frac{1}{N}\sum_{i=1}^{N}\sum_{t'=T+1}^{T+H}(\hat{\mathbf{Y}}_{n,t'}^{(\mathrm{S})} - \mathbf{Y}_{n,t'})^2.\end{aligned} \tag{5}$$

Here, $\mathcal{L}^{(\mathrm{T})}$ represents the loss of the GNN-based teacher model, denoted by the superscript (T). Similarly, $\mathcal{L}^{(\mathrm{S})}$ represents the loss of the MLP-based student model, denoted by the superscript (S). N corresponds to the number of sensing regions within the spatio-temporal graph $\mathcal{G}$, and H denotes the number of predicted time steps. $\hat{\mathbf{Y}}_{n,t'}^{(\mathrm{T})}$ represents the predicted traffic volume of node n at time step t' by the teacher model, while $\hat{\mathbf{Y}}_{n,t'}^{(\mathrm{S})}$ represents the predicted traffic volume of the same node and time step by the student model. $\mathbf{Y}_{n,t'}$ denotes the ground truth traffic volume of node n at time step t'. To further measure the distribution difference of the GNN-based teacher model and that of the MLP-based student model, we adopt Kullback Leibler (KL) (Hu and Hong 2013) divergence into the distribution alignment loss $\mathcal{L}^{(\mathrm{KL})}$:

$$\mathcal{L}^{(\mathrm{KL})} = \sum_{n=1}^{N}\sum_{t'=T+1}^{T+H}\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})} \cdot \log(\mathrm{Softmax}(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})})). \tag{6}$$

Here, $\mathcal{L}^{(\mathrm{KL})}$ represents the Kullback-Leibler (KL) divergence loss, denoted by the superscript (KL). $\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})}$ denotes the output of the Temporal Convolutional Network (TCN) for node n at time step t' from the teacher model, while $\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})}$ represents the corresponding output from the student model. Minimizing $\mathcal{L}^{(\mathrm{KL})}$ aims to align the predicted probability distribution of the student model with that of the teacher model. This process effectively incorporates spatio-temporal knowledge from the teacher into the student's predictions.

Distribution Alignment. To further transfer spatio-temporal knowledge from the representation space, we introduce distribution alignment. Our model aims to achieve consistency in the embeddings of the same region from both the teacher and student models. To account for variations in the spatio-temporal context within the latent semantic space, we assign different consistency strengths to embeddings of different region pairs. This is achieved using a similarity function denoted by $\eta(\cdot)$, where we leverage embedding cosine similarity.

Our approach, which involves distillation from both the spatial (with loss $\mathcal{L}^{(\mathrm{P})}$) and temporal (with loss $\mathcal{L}^{(\mathrm{E})}$) dimensions using contrastive learning, is formally presented as follows:

$$\begin{aligned}\mathcal{L}^{(\mathrm{P})} &= \sum_{n=1}^{N}\sum_{t'=T+1}^{T+H} -\log\frac{\exp(\frac{\eta(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})},\mathbf{E}_{n,t'}^{(\mathrm{T})})}{\tau_2})}{\sum_{n'\neq n}\exp(\frac{\eta(\tilde{\mathbf{E}}_{n',t'}^{(\mathrm{S})},\mathbf{E}_{n,t'}^{(\mathrm{T})})}{\tau_2})},\\ \mathcal{L}^{(\mathrm{E})} &= \sum_{n=1}^{N}\sum_{t'=T+1}^{T+H} -\log\frac{\exp(\frac{\eta(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})},\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})})}{\tau_3})}{\sum_{n'\neq n}\exp(\frac{\eta(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})},\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})})}{\tau_3})}.\end{aligned} \tag{7}$$

The parameters τ_2 and τ_3 control the temperature of the softmax function used in the contrastive loss during training. $\mathcal{L}^{(\mathrm{P})}$ and $\mathcal{L}^{(\mathrm{E})}$ represent the spatial loss and temporal loss, respectively, denoted by the superscripts (P) and (E). $\mathbf{E}_{n,t'}^{(\mathrm{T})}$ represents the output embedding from the GCN layers of the teacher model for region n at time step t'. Similarly, $\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})}$ represents the output embedding from the TCN layers of the teacher model for the same region and time step. Finally, $\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})}$ denotes the output embedding from TCN layers of the student model for region n at time step t'.

Model Optimization. Following the learning paradigm of knowledge distillation, we first train the GNN-based teacher model of LightST until convergence using the loss function $\mathcal{L}^{(\mathrm{T})}$ from Equation 5. This involves feeding mini-batches of traffic observation tensors into the model and optimizing it. We then perform joint training to optimize both the MLP-based student model and the teacher model together. The overall objective function is an integration of the optimized objectives, which is shown as follows:

$$\mathcal{L} = \mathcal{L}^{(\mathrm{S})} + \lambda_1 \cdot \mathcal{L}^{(\mathrm{KL})} + \lambda_2 \cdot (\mathcal{L}^{(\mathrm{P})} + \mathcal{L}^{(\mathrm{E})}), \tag{8}$$

where λ_1 and λ_2 are loss weights. The training process of our LightST is elaborated in Algorithm 1 in Appendix A.1.

Discussion of Model

Model Complexity. To evaluate the efficiency improvement of the proposed MLP-based student model, we analyzed its time complexity in comparison to the GNN-based teacher model. The teacher model has a higher time complexity due to its graph information propagation in the encoder, which requires $\mathcal{O}(|\mathcal{E}| \times L \times d)$, where $|\mathcal{E}|$ is the number of edges, and L is the number of graph layers. In contrast, the proposed student model only requires $\mathcal{O}(B \times L' \times d^2)$ for CL-enhanced distribution alignment, where B is the number of samples in each batch, and L' is the number of MLP layers. In summary, our analysis shows that the proposed MLP-based student model is significantly more efficient than the GNN-based traffic prediction methods, making it a promising framework for large-scale traffic data in practical spatio-temporal data mining scenarios.

Model Theoretical Analysis. We discuss how adaptive spatio-temporal distillation can alleviate the over-smoothing effects in spatio-temporal GNNs. We will begin by introducing the message passing schema that is used to propagate information along the graph-structured path in our spatial graph $\mathcal{G}$:

$$\mathbf{E}_n^{(\mathrm{T})(L)} = \sum_{j \in \mathcal{N}_n} \Big(\sum_{\mathcal{Z}_{n,j}^L} \prod_{n_k, n_h \in \mathcal{Z}_{n,j}^L} \frac{1}{\sqrt{d_k d_h}} \Big) \cdot \mathbf{E}_j^{(\mathrm{T})(0)}, \quad (9)$$

where $\mathcal{Z}_{n,j}^L$ represents the maximum length L of a possible path from the n-th region node and the j-th region node, with the intermediate connection nodes k and h. The variables d_k and d_h refer to the degrees of these intermediate nodes. From the above Eq 9, it is important to note that the weight of $\mathbf{E}_j^{(\mathrm{T})(0)}$ is non-learnable, which means it cannot be adjusted during the message passing process over noisy graph structures when generating the region embedding $\mathbf{E}_n^{(\mathrm{T})(L)}$. $\mathcal{N}_n$ denotes the set of neighbour nodes of region n. Our framework provides a solution to this issue by introducing a learnable and adaptive knowledge distillation approach. Here, we analyze the gradients of our knowledge distillation with KL divergence alignment objective $\mathcal{L}^{(\mathrm{KL})}$ given the corresponding embeddings $\tilde{\mathbf{E}}_n^{(\mathrm{S})}$ of region n via the student model as follows:

$$\frac{\partial \mathcal{L}^{(\mathrm{KL})}}{\partial \tilde{\mathbf{E}}_n^{(\mathrm{S})}} = \sum_{n=1}^{N} \sum_{t'=T+1}^{T+H} \omega \cdot \frac{\partial(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})}, \tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})})}{\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})}},$$
$$\omega = \frac{1}{\mathrm{softmax}(\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})})} \Big(-\frac{e^{\tilde{\mathbf{E}}_{j,t'}^{(\mathrm{S})}}}{\sum_{n'=1}^{N} e^{\tilde{\mathbf{E}}_{n',t'}^{(\mathrm{S})}}}\Big)\Big(-\frac{e^{\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{S})}}}{\sum_{n'=1}^{N} e^{\tilde{\mathbf{E}}_{n',t'}^{(\mathrm{S})}}}\Big). \quad (10)$$

Here, $\tilde{\mathbf{E}}_{n',t'}^{(\mathrm{S})}$ represents the region embedding obtained from the TCN layers of the student model, denoted by the superscript (S). The subscripts n', t' indicate node n' at time step t'. Similarly, $\tilde{\mathbf{E}}_{n,t'}^{(\mathrm{T})}$ represents the region embedding of node n at time step t' obtained from the TCN layers of the teacher model. The derivations in Equation 10 demonstrate that the region embeddings are refined through the transferred knowledge from the teacher model using the derived weight ω. While recursive message passing can lead to oversmoothing of the representations, our framework automatically adapts the knowledge transfer process, mitigating these over-smoothing effects.

Evaluation

Experimental Setting

Datasets. In this study, we conduct a series of experiments using real-life traffic flow datasets from California, specifically the PEMS3, PEMS4, PEMS7, PEMS8 and PeMS-Bay datasets released by (Song et al. 2020). The traffic data is aggregated into 5-minute time intervals, resulting in 12 points of data per hour. Additionally, we construct the spatial graph of our traffic sensing regions based on the road network, which models the relationships between traffic flow patterns in different regions of the city.

Baselines. We conduct a comprehensive evaluation of LightST by comparing it against 18 baselines, including many state-of-the-art GNN-based traffic prediction models, which are categorized into six groups: 1) **GCN-based methods**: STGCN (Yu, Yin, and Zhu 2018),DCRNN (Li et al. 2018), GWN (Shleifer, McCreery, and Chitters 2019); 2) **GAT-based approaches**: ASTGCN (Zhu et al. 2021), LSTGCN (Han and Gong 2022), DSTAGNN (Lan et al. 2022); 3) **Differential GNNs**: STG-ODE (Fang et al. 2021); 4) **GNNs enhanced with Zigzag Persistence**: Z-GCNETs (Chen, Segovia, and Gel 2021) and TAMP (Chen, Segovia-Dominguez et al. 2021); 5) **Hybrid spatio-temporal GNNs**: FOGS (Rao et al. 2022), AGCRN (Bai, Yao et al. 2020), STSGCN (Song et al. 2020), STFGNN (Li and Zhu 2021). 6) **Distillation methods**: KD-Pruning (Izadi, Safayani, and Mirzaei 2024) and STKD (Wang et al. 2024).

Effectiveness Evaluation

We evaluate the effectiveness of our method, LightST, and the baselines on 5 datasets in terms of MAE, MAPE, and RMSE metrics, as shown in Table 1. Based on results, we have following observations:

Superior Prediction Accuracy. Our proposed method has consistently outperforms other baselines across all four datasets, in most evaluation cases. This can be attributed to several key factors that contribute to the effectiveness of our approach. *Firstly*, we are able to successfully distill spatial and temporal dynamics from the teacher model. This enables the student to capture time-evolving traffic dependencies across geographical regions and time slots without relying on cumbersome message passing frameworks. *Secondly*, our adaptive knowledge distillation method, realized through our dual-level knowledge transfer, guides student learning with appropriate knowledge to alleviate the over-smoothing effects of the GNN architecture. *Third*, by enabling cross-region and cross-time dependency modeling in an adaptive manner, our spatio-temporal knowledge distillation alleviates the effects of noisy adjacent relationships, contributing to the robustness of traffic flow prediction.

Performance Comparison among Baselines. Among the various baselines, we observe that methods such as STSGCN and STFGNN, which incorporate time-aware spatial dependency, perform better than approaches such as STGCN and DCRNN, which only consider stationary spatial correlations among regions. This highlights the importance of capturing temporal dynamics when encoding spatial dependency relationships among regions. In contrast to distillation methods like STKD and KD-pruning, which incorporate LSTM or GNNs into student models, potentially leading to over-smoothing and suboptimal performance, TCNs demonstrate superior efficacy in capturing temporal dynamics compared to LSTMs. Our proposed spatio-temporal knowledge distillation paradigm is designed to transfer time-aware spatial dependency knowledge from the teacher model to the MLP student model. By doing so, the student model is able to capture complex spatio-temporal patterns of traffic flow, resulting in state-of-the-art traffic prediction performance.

Model Scalability Investigation

To evaluate the efficiency of our proposed LightST, we conduct experiments on the large PeMSD7 dataset competing

Models	PeMS-Bay			PeMSD4			PeMSD8			PeMSD3			PeMSD7		
	MAE ↓	RMSE ↓	MAPE ↓	MAE ↓	RMSE ↓	MPAE ↓	MAE ↓	RMSE ↓	MPAE ↓	MAE ↓	RMSE ↓	MAPE ↓	MAE ↓	RMSE ↓	MAPE ↓
HA	2.88	5.59	6.82%	38.03	59.24	27.88%	34.86	52.04	24.07%	31.58	52.39	33.78%	45.12	65.64	24.51%
VAR	2.32	5.25	5.61%	24.54	38.61	17.24%	19.19	29.80	13.10%	23.65	38.26	24.51%	50.22	75.63	32.22%
DSANet	2.16	4.97	5.54%	22.79	35.77	17.12%	17.14	26.96	11.32%	21.29	34.55	23.21%	31.36	49.11	14.43%
DCRNN	2.07	4.74	4.90%	24.70	38.12	14.17%	17.86	27.83	11.45%	17.99	30.31	18.34%	25.22	38.61	11.82%
STGCN	2.42	5.33	5.58%	22.70	35.55	14.59%	18.02	27.83	11.40%	17.55	30.42	17.34%	25.33	39.34	11.21%
GWN	1.95	4.52	4.63%	25.45	39.70	17.29%	19.13	31.05	12.68%	19.12	32.77	18.89%	26.39	41.50	11.97%
ASTGCN	2.10	4.77	5.30%	22.93	35.22	16.56%	18.25	28.06	11.64%	17.34	29.56	17.21%	24.01	37.87	10.73%
LSGCN	2.13	4.82	5.18%	21.53	33.86	13.18%	17.73	26.76	11.30%	17.94	29.85	16.98%	27.31	41.16	11.98%
STSGCN	2.10	4.74	5.28%	21.19	33.65	13.90%	17.13	26.86	10.96%	17.48	29.21	16.78%	24.26	39.03	10.21%
AGCRN	1.96	4.57	4.69%	19.83	32.26	12.97%	15.95	25.22	10.09%	15.98	28.25	15.23%	22.37	36.55	9.12%
STFGNN	1.83	4.33	4.19%	19.83	31.88	13.02%	16.64	26.22	10.60%	16.77	28.34	16.30%	22.07	35.80	9.21%
STGODE	2.02	4.40	4.72%	20.84	32.82	13.77%	16.81	25.97	10.62%	16.50	27.84	16.69%	22.99	37.54	10.14%
Z-GCNETs	2.03	4.38	4.71%	19.50	31.61	12.78%	15.76	25.11	10.01%	16.64	28.15	16.39%	21.77	35.17	9.25%
TAMP	2.04	4.45	4.76%	19.74	31.74	13.22%	16.36	25.98	10.15%	16.46	28.44	15.37%	21.84	35.42	9.24%
FOGS	2.07	4.51	4.80%	19.74	31.66	13.05%	15.73	24.92	9.88%	15.89	25.74	15.13%	21.28	34.88	**8.95%**
DSTAGNN	2.13	4.79	5.32%	19.30	31.46	12.72%	15.67	24.77	9.94%	15.57	27.21	14.68%	21.42	34.51	9.01%
STKD	2.08	4.56	4.82%	19.86	31.93	13.18%	15.81	25.07	10.02%	16.03	25.95	15.76%	21.64	34.96	9.03%
KD-Pruning	2.23	4.97	5.34%	21.22	34.63	14.15%	17.46	27.09	11.74%	17.12	29.87	17.06%	24.55	38.17	11.90%
LightST(Ours)	**1.78**	**3.88**	**4.15%**	**19.21**	**31.31**	**12.70%**	**15.43**	**24.52**	**9.84%**	**15.11**	**24.74**	**14.41%**	**20.78**	**33.95**	8.98%

Table 1: Overall performance of traffic prediction on PeMS-Bay, PeMSD4, PeMSD8, PeMSD3 and PeMSD7

with several state-of-the-art baselines. We conduct the experiments on a server with 10 cores of Intel(R) Core(TM) i9-9820X CPU @ 3.30GHz, 64.0GB RAM, and 4 Nvidia GeForce RTX 3090 GPU. The results for inference time and forecasting accuracy are shown in Table 2. Our analysis yields two key observations. First, our LightST achieves competitive performance in terms of accuracy metrics, *i.e.*, MAE, MAPE, and RMSE. This is particularly noteworthy given the potential for over-smoothing on large-scale spatial region graphs, which our framework avoids by not explicitly introducing graph message passing and instead distilling denoised spatio-temporal knowledge into graph-less designations. Second, our LightST achieves much faster inference time than the compared baseline models, which is attributed to the fact that LightST does not require recursive graph-based information propagation operations during inference phase. While our traffic flow predictor is a simple graph-less neural network, its achieved superior performance suggests the effectiveness of our knowledge distillation paradigm in injecting complex global spatio-temporal dependencies across high-order region and time connections into the student. The ability to achieve high accuracy with fast inference time is particularly important in practical applications, where traffic forecasting models need to operate in real-time urban sensing.

Ablation Study and Effectiveness Analyses

To assess the impact of each component in our knowledge distillation framework on prediction results and speed, we conducted an ablation study across four traffic datasets using model variants. These variants include: 1) *w/o E-KD*, which disables embedding-level knowledge distillation for transferring spatio-temporal signals from the latent representation space; 2) *w/o E-S*, which omits adaptive embedding alignment with spatial information by removing $\mathcal{L}^{(\mathrm{P})}$ from the joint loss $\mathcal{L}$; and 3) *w/o E-T*, which excludes $\mathcal{L}^{(\mathrm{E})}$ from $\mathcal{L}$ to avoid capturing temporal information during embedding-level knowledge distillation. The results, presented in Figure 3, reveal key observations. Firstly, the *w/o E-KD* variant performs significantly worse than our full model, highlighting the crucial role of embedding-level knowledge distillation in transferring spatio-temporal signals. Notably, KL divergence is the most computationally demanding component of our model. Secondly, the superior performance of our model compared to *w/o E-S* and *w/o E-T* demonstrates the effectiveness of adaptive embedding alignment across both spatial and temporal domains in capturing complex cross-location and cross-time traffic dependencies.

Datasets	PeMSD7				
Method	MAE ↓	RMSE ↓	MAPE ↓	Inference ↓	Faster x ↑
ASTGCN	24.01	37.87	10.73%	20.06s	**7.99** ×
STFGNN	22.07	35.80	9.21%	53.81s	**21.44** ×
STGODE	22.99	37.54	10.14%	24.79s	**9.88** ×
DSTAGNN	21.42	34.51	9.01%	110.06s	**43.85** ×
Ours	**20.78**	**33.95**	8.98%	**2.51s**	-
Datasets	PeMS-Bay				
Method	MAE ↓	RMSE ↓	MAPE ↓	Inference ↓	Faster x ↑
ASTGCN	2.10	4.77	5.30	40.08s	**7.82** ×
STFGNN	1.83	4.33	4.19%	98.18s	**19.18** ×
STGODE	2.02	4.40	4.72%	208.51s	**40.72** ×
DSTAGNN	2.13	4.79	5.32%	86.97s	**16.99** ×
Ours	**1.78**	**3.88**	**4.15%**	**5.12s**	-

Table 2: Model Efficiency Study

Hyperparameter Study

The aim of this section is to evaluate the effects of key hyperparameters on the performance of our framework, LightST.

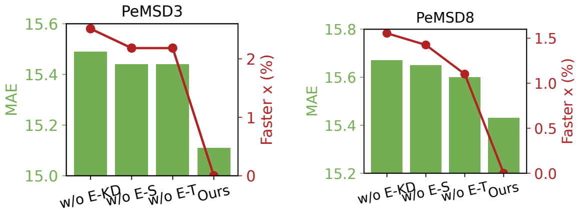

Figure 3: Ablation study of sub-modules in our spatio-temporal knowledge distillation paradigm.

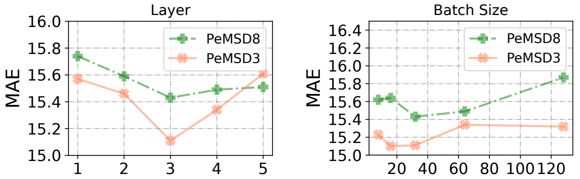

Figure 4: Hyparameter study on PeMSD8 and PeMSD3 in terms of MAE.

We present our results on PeMSD8 and PeMSD3 datasets in terms of MAE and RMSE in Figure 4. We summarie our observations as follows: 1) Figure 4 show the effect of the number of MLP layers (ranging from $\{1, 2, 3, 4, 5\}$) and varying batch size (ranging from $\{2^3, 2^4, 2^5, 2^6, 2^7\}$) on performance. Our framework, LightST, achieves the best performance on PeMSD8 and PeMSD3 when the number of layers is 3 and the batch size is 32. Even when LightST achieves the worst performance, it still outperforms most of the baselines. These results suggest that the performance of our LightST is not sensitive to the MLP depth and batch size. 2) λ_1, λ_2 serve as loss weights to control how strongly our prediction-level and embedding-level restrict the joint model training. Figure 4 show that λ_1 and λ_2 jointly affect the strength of the optimization of knowledge distillation. We find that a larger weight of distillation causes performance maintenance, enabling MLP to learn sufficient knowledge.

Related Work

Traffic Flow Prediction. Numerous neural network architectures have been proposed for traffic prediction, including convolutional neural networks (CNNs) (Zhang, Zheng, and Qi 2017), recurrent neural networks (RNNs) (Lv et al. 2018), attention mechanisms (Yao et al. 2019), and graph neural networks (GNNs) (Li and Zhu 2021). CNNs have proven effective in modeling citywide traffic maps as images for spatio-temporal pattern encoding (Diao et al. 2019; Zhang, Zheng, and Qi 2017), while RNNs excel at capturing temporal dependencies in time-evolving traffic data (Lv et al. 2018). To model spatial traffic similarities adaptively, research has explored spatial dependency graphs with learnable region adjacency matrices (Shleifer, McCreery, and Chitters 2019; Bai, Yao et al. 2020; Rao et al. 2022; Lan et al. 2022). For instance, DSTAGNN (Lan et al. 2022) utilizes a multi-head attention mechanism to exploit spatial correlations with multi-scale neighborhoods. FOGS (Rao et al. 2022) learns a spatial-temporal correlation graph using first-order gradients during training. While GNN-enhanced traffic prediction models hold promise, their computational complexity hinders scalability and real-world deployment. This study addresses this challenge by leveraging spatio-temporal knowledge distillation to reduce inference time, enabling our model to effectively scale to larger datasets.

Recent traffic prediction methods have employed Multilayer Perceptron (MLP)-based networks, including STID (Shao et al. 2022) and ST-MLP (Wang et al. 2023). STID leverages MLPs to capture spatio-temporal dynamics using spatial and temporal identity mappings. ST-MLP (Wang et al. 2023) focuses on using MLPs exclusively, relying on predefined spatial graph structures to improve efficiency. While Transformer-based approaches, such as STAEformer (Liu et al. 2023), show promising performance, their high computational cost and resource requirements hinder their efficiency and scalability.

Conclusion

We propose LightST, a scalable and high-performance traffic flow prediction framework that offers both model efficiency and generalization capability, which are often lacking in existing solutions. We draw inspiration from the knowledge distillation paradigm to achieve efficiency and retain the awareness of high-order spatial-temporal traffic dependencies across locations and time. To this end, we perform both explicit prediction-level and implicit embedding-level distillation to transfer spatio-temporal knowledge from a cumbersome GNN teacher to a simple yet effective MLP student. Additionally, our model adopts a new adaptive model alignment schema to further enhance the student model by alleviating over-smoothing effects. In future work, we plan to enhance our spatio-temporal knowledge distillation with causal inference, to identify confounding variables that may affect the spatio-temporal data and adjust for their effects in the knowledge distillation process.

Appendices

Algorithm of LightST

More Related Work Knowledge Distillation for Graphs. Knowledge distillation on graphs provides a promising approach for transferring knowledge from complex teacher GNNs to smaller student models, effectively reducing computational costs while preserving accuracy (Guo et al. 2022; Wu, Lin et al. 2022; Feng et al. 2022; Qin et al. 2021). This technique has been applied to various graph applications, including node/graph classification (Zhang et al. 2022; He et al. 2022), 2022], social media analysis (Qian et al. 2021), and recommender systems (Tao et al. 2022). Adversarial training, involving a discriminator and generator, has been employed to enhance knowledge distillation (He et al. 2022). Qian et al. (2021) applied knowledge distillation to a heterogeneous graph for analyzing drug trafficking from social media data. Tao et al. (2022) proposed a distillation-enhanced relational encoder to improve recommendation accuracy by capturing user-item interactions and social connections. STKD (Wang et al. 2024) is a method that adopts distillation via 1D CNN

Algorithm 1: The LightST Learning Algorithm

```
Input: Historical observation tensor $\mathbf{X} \in \mathbb{R}^{N \times T}$ and
       the spatial graph $\mathcal{G}$, $\tau_2$, $\tau_3$, $\lambda_1$,$\lambda_2$, learning
       rate $\eta$, maximum training epochs $S$
Output: trained parameters in $\Theta$
Initialize all parameters in $\Theta$;
Train teacher model by Equation 5, and collect the
  traffic embeddings at each time interval $t$, denoted
  as $E_t^{(\mathrm{T})}$
for $epoch = 1, 2, ..., S$ do
    Calculate the loss $\mathcal{L}^{(\mathrm{S})}$ by Equation 5;
    Perform prediction-level distillation and compute
      the KL loss $\mathcal{L}^{(\mathrm{KL})}$ by Equation 6;
    Perform embedding-level distillation from spatial
      and temporal dimensions by Equation 7;
    Calculate the overall loss $\mathcal{L}$ by Equation 8;
    for Optimizing parameters on weight factor in
      prediction-level do
        Calculating weight factor according to Eq 10
    end
    for $\theta \in \Theta$ do
        $\theta = \theta - \eta \cdot \frac{\partial \mathcal{L}^{(\mathrm{S})}}{\partial \theta}$
    end
end
Return all parameters $\Theta$
```

Datasets	#Sensors	Time Period	Time Steps	Interval
PeMSD4	307	2018/1/1-2018/2/28	16,992	5 minutes
PeMSD8	170	2016/7/1-2016/8/31	17,856	5 minutes
PeMSD3	358	2018/9/1-201811/30	26,208	5 minutes
PeMSD7	883	2017/5/1-2017/8/31	28,224	5 minutes
PeMS-Bay	325	2017/1/1-2017/5/31	52,116	5 minutes

Table 3: Data Description and Statistics.

and LSTM to detect traffic anomalies. However, 1D CNNs are limited by fixed-size windows, hampering their ability to capture long-range dependencies crucial for modeling complex temporal patterns in traffic data.

Acknowledgements

This project is supported by HKU-SCF FinTech Academy and Shenzhen-Hong Kong-Macao Science and Technology Plan Project (Category C Project: SGDX202108 23103537030) and Theme-based Research Scheme T35-710/20-R. The Australian Research Council partially supports this work under the streams of Future Fellowship (Grant No. FT210100624), and the Linkage Project (Grant No. LP230200892). We also appreciate the help of Dr.Huang, Dr.Xia, Dr.Li and Dr.Tang.

References

Bai, L.; Yao, L.; et al. 2020. Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting. In *International Conference on Neural Information Processing Systems (NeurIPS)*.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *International Conference on Machine Learning (ICML)*, 1725–1735. PMLR.

Chen, Y.; Segovia, I.; and Gel, Y. R. 2021. Z-GCNETs: time zigzags at graph convolutional networks for time series forecasting. In *International Conference on Machine Learning (ICML)*, 1684–1694. PMLR.

Chen, Y.; Segovia-Dominguez, I.; et al. 2021. TAMP-S2GCNets: coupling time-aware multipersistence knowledge representation with spatio-supra graph convolutional networks for time-series forecasting. In *International Conference on Learning Representations (ICLR)*.

Diao, Z.; Wang, X.; Zhang, D.; Liu, Y.; Xie, K.; and He, S. 2019. Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, 890–897.

Fang, Z.; Long, Q.; Song, G.; and Xie, K. 2021. Spatial-Temporal Graph ODE Networks for Traffic Flow Forecasting. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 364–373. ACM.

Feng, K.; Li, C.; Yuan, Y.; and Wang, G. 2022. FreeKD: Free-direction Knowledge Distillation for Graph Neural Networks. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 357–366.

Gao, X.; Chen, T.; Zang, Y.; Zhang, W.; Nguyen, Q. V. H.; Zheng, K.; and Yin, H. 2024a. Graph condensation for inductive node representation learning. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, 3056–3069. IEEE.

Gao, X.; Chen, T.; Zhang, W.; Li, Y.; Sun, X.; and Yin, H. 2024b. Graph condensation for open-world graph learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 851–862.

Gao, X.; Yu, J.; Chen, T.; Ye, G.; Zhang, W.; and Yin, H. 2024c. Graph condensation: A survey. *arXiv preprint arXiv:2401.11720*.

Guo, Z.; Shiao, W.; Zhang, S.; Liu, Y.; Chawla, N.; Shah, N.; and Zhao, T. 2022. Linkless Link Prediction via Relational Distillation. In *International Conference on Machine Learning (ICML)*.

Han, X.; and Gong, S. 2022. LST-GCN: Long Short-Term Memory Embedded Graph Convolution Network for Traffic Flow Forecasting. *Electronics*, 11(14): 2230.

He, H.; Wang, J.; Zhang, Z.; and Wu, F. 2022. Compressing deep graph neural networks via adversarial knowledge distillation. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 534–544.

Hu, Z.; and Hong, L. J. 2013. Kullback-Leibler divergence constrained distributionally robust optimization. *Available at Optimization Online*, 1(2): 9.

Izadi, M.; Safayani, M.; and Mirzaei, A. 2024. Knowledge Distillation on Spatial-Temporal Graph Convolutional Network for Traffic Prediction. *arXiv preprint arXiv:2401.11798*.

Lan, S.; Ma, Y.; Huang, W.; Wang, W.; Yang, H.; and Li, P. 2022. Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In *International Conference on Machine Learning (ICML)*, 11906–11917. PMLR.

Li, M.; and Zhu, Z. 2021. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, 4189–4196.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations (ICLR)*. OpenReview.net.

Liang, Y.; Ouyang, K.; Jing, L.; Ruan, S.; Liu, Y.; Zhang, J.; Rosenblum, D. S.; and Zheng, Y. 2019. Urbanfm: Inferring fine-grained urban flows. In *International Conference on Knowledge Discovery & Data Mining (KDD)*, 3132–3142.

Liu, H.; Dong, Z.; Jiang, R.; Deng, J.; Deng, J.; Chen, Q.; and Song, X. 2023. Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, 4125–4129.

Lv, Z.; Xu, J.; Zheng, K.; Yin, H.; Zhao, P.; and Zhou, X. 2018. Lc-rnn: A deep learning model for traffic speed prediction. In *International Joint Conference on Artificial Intelligence (IJCAI)*, volume 2018, 27.

Pan, Z.; Liang, Y.; Wang, W.; Yu, Y.; Zheng, Y.; and Zhang, J. 2019. Urban traffic prediction from spatio-temporal data using deep meta learning. In *International Conference on Knowledge Discovery & Data Mining (KDD)*, 1720–1730.

Qian, Y.; Zhang, Y.; Ye, Y.; Zhang, C.; et al. 2021. Distilling meta knowledge on heterogeneous graph for illicit drug trafficker detection on social media. *Advances in Neural Information Processing Systems (NeurIPS)*, 34: 26911–26923.

Qin, C.; Zhao, H.; Wang, L.; Wang, H.; Zhang, Y.; and Fu, Y. 2021. Slow learning and fast inference: Efficient graph similarity computation via knowledge distillation. *Advances in Neural Information Processing Systems (NeurIPS)*, 34: 14110–14121.

Rao, X.; Wang, H.; Zhang, L.; Li, J.; Shang, S.; and Han, P. 2022. Fogs: First-order gradient supervision with learning-based graph for traffic flow forecasting. In *International Joint Conference on Artificial Intelligence (IJCAI)*. ijcai.org.

Shao, Z.; Zhang, Z.; Wang, F.; Wei, W.; and Xu, Y. 2022. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 4454–4458.

Shleifer, S.; McCreery, C.; and Chitters, V. 2019. Incrementally improving graph WaveNet performance on traffic prediction. *arXiv preprint arXiv:1912.07390*.

Song, C.; Lin, Y.; Guo, S.; and Wan, H. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, 914–921.

Tao, Y.; Li, Y.; Zhang, S.; et al. 2022. Revisiting Graph based Social Recommendation: A Distillation Enhanced Social Graph Network. In *ACM Web Conference (WWW)*, 2830–2838.

Wang, X.; Ma, Y.; Wang, Y.; Jin, W.; Wang, X.; Tang, J.; et al. 2020. Traffic flow prediction via spatial temporal graph neural network. In *The Web Conference (WWW)*, 1082–1092.

Wang, X.; Wang, Z.; Wang, E.; and Sun, Z. 2024. Spatial-temporal knowledge distillation for lightweight network traffic anomaly detection. *Computers & Security*, 137: 103636.

Wang, Z.; Nie, Y.; Sun, P.; Nguyen, N. H.; Mulvey, J.; and Poor, H. V. 2023. St-mlp: A cascaded spatio-temporal linear framework with channel-independence strategy for traffic forecasting. *arXiv preprint arXiv:2308.07496*.

Wu, L.; Lin, H.; et al. 2022. Knowledge Distillation Improves Graph Structure Augmentation for Graph Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; and Zhang, C. 2020. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *International Conference on Knowledge Discovery & Data Mining (KDD)*, 753–763.

Yao, H.; Tang, X.; Wei, H.; Zheng, G.; and Li, Z. 2019. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, 5668–5675.

Yu, B.; Yin, H.; and Zhu, Z. 2018. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 3634–3640. ijcai.org.

Zhang, J.; Zheng, Y.; and Qi, D. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Zhang, Q.; Wang, H.; Long, C.; Su, L.; He, X.; Chang, J.; Wu, T.; Yin, H.; Yiu, S.-M.; Tian, Q.; et al. 2024. A Survey of Generative Techniques for Spatial-Temporal Data Mining. *arXiv preprint arXiv:2405.09592*.

Zhang, S.; Liu, Y.; Sun, Y.; and Shah, N. 2022. Graph-less neural networks: Teaching old mlps new tricks via distillation. In *International Conference on Learning Representations (ICLR)*.

Zhou, K.; Huang, X.; Li, Y.; Zha, D.; Chen, R.; and Hu, X. 2020. Towards deeper graph neural networks with differentiable group normalization. *Advances in Neural Information Processing Systems (NeurIPS)*, 33: 4917–4928.

Zhu, J.; Wang, Q.; Tao, C.; Deng, H.; et al. 2021. AST-GCN: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting. *IEEE Access*, 9: 35973–35983.