Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting [Review]

박태준

March 3, 2023

Abstract

- Timely accurate traffic forecast is crucial for urban traffic control and guidance.
- In this paper, we propose a novel deep learning framework, **Spatio-Temporal Graph Convolutional Networks (STGCN)**.
- Instead of applying regular convolutional and recurrent units, we formulate the problem on graphs and build the model with complete convolutional structures, which enable much faster training speed with fewer parameters.

1 Introduction

- To take full advantage of spatial features, some researchers use convolutional neural network (CNN) to capture adjacent relations among the traffic network, along with employing recurrent neural network (RNN) on time axis.
- We introduce several strategies to effectively model temporal dynamics and spatial dependencies.
- It is the first time that to apply purely convolutional structures to extract spatio-temporal features simultaneously from graph-structured time series in a traffic study.

2 Preliminary

2.1 Traffic Prediction on Road Graphs

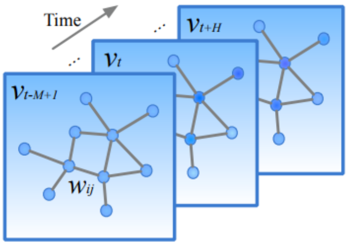

Figure 1: Graph-structured traffic data. Each v_t indicates a frame of current traffic status at time step t, which is recorded in a graph-structured data matrix.

- In this work, we define the traffic network on a graph and focus on structured traffic time series.
- The observation v_t is not independent but linked by pairwise connection in graph.
- Therefore, the data point v_t can be regarded as a graph signal that is defined on an undirected graph (or directed one) $\mathcal{G}$ with weights $w_{i,j}$ as shown in Figure 1.
- At the t-th time step, in graph

$$\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}, W),$$

  ▷ $\mathcal{V}_t$ is a finite set of vertices, corresponding to the observations from n monitor stations in traffic network;

  ▷ $\mathcal{E}$ is a set of edges, indicating the connectedness between stations;

  ▷ $W \in \mathbb{R}^{n \times n}$ denotes the weighted adjacency matrix of $\mathcal{G}_t$.

2.2 Convolutions on Graphs

- A standard convolution for regular grids is clearly not applicable to general graphs.

- There are two basic approaches currently exploring how to generalize CNNs to structured data forms.
  - ▷ One is to expand the spatial definition of a convolution [Niepert et al., 2016],
    - └ The former approach rearranges the vertices into certain grid forms which can be processed by normal convolutional operations.
  - ▷ The other is to manipulate in the spectral domain with graph Fourier transforms [Bruna et al., 2013].
    - └ The latter one introduces the spectral framework to apply convolutions in spectral domains, often named as the spectral graph convolution.
- We introduce the notion of graph convolution operator "$*_{\mathcal{G}}$" based on the conception of spectral graph convolution, as the multiplication of a signal $x \in \mathbb{R}^n$ with a kernel Θ,

$$\begin{aligned} \Theta *_{\mathcal{G}} x &= \Theta(L)x \\ &= \Theta(U\Lambda U^\top)x \\ &= U\Theta(\Lambda)U^\top x, \end{aligned} \tag{2}$$

  where graph Fourier basis $U \in \mathbb{R}^{n\times n}$ is the matrix of eigenvectors of the normalized graph Laplacian

$$L = I_n - D^{-\frac{1}{2}}WD^{-\frac{1}{2}} = U\Lambda U^\top \in \mathbb{R}^{n\times n}$$

  - ▷ I_n is an identity matrix,
  - ▷ $D \in \mathbb{R}^{n\times n}$ is the diagonal degree matrix with $D_{ii} = \sum_j W_{ij}$,
  - ▷ $\Lambda \in \mathbb{R}^{n\times n}$ is the diagonal matrix of eigenvalues of L,
  - ▷ filter $\Theta(\Lambda)$ is also a diagonal matrix.
- By this definition, a graph signal x is filtered by a kernel Θ with multiplication between Θ and graph Fourier transform $U^\top x$.

3 Proposed Model

3.1 Network Architecture

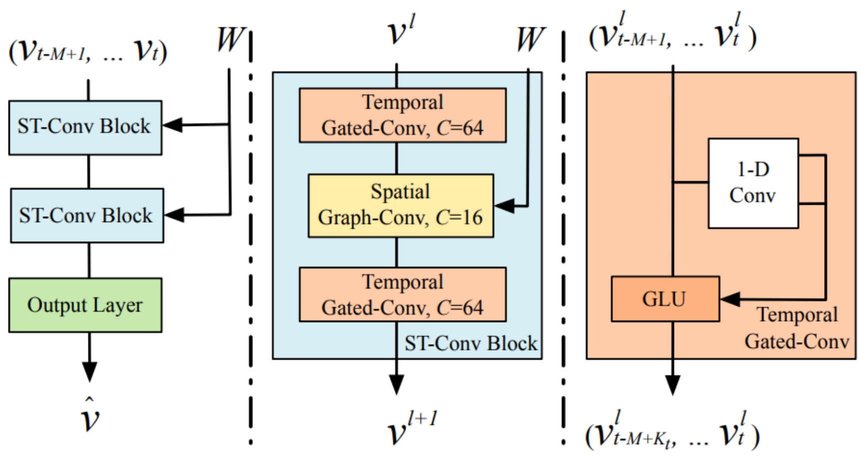

Figure 2: Architecture of spatio-temporal graph convolutional networks. The framework STGCN consists of two spatio-temporal convolutional blocks (ST-Conv blocks) and a fully-connected output layer in the end. Each ST-Conv block contains two temporal gated convolution layers and one spatial graph convolution layer in the middle. The residual connection and bottleneck strategy are applied inside each block. The input $v_{t-M+1}, \ldots, v_t$ is uniformly processed by ST-Conv blocks to explore spatial and temporal dependencies coherently. Comprehensive features are integrated by an output layer to generate the final prediction $\hat{v}$.

- In this section, we elaborate on the proposed architecture of **spatio-temporal graph convolutional networks (STGCN)**.
- As shown in Figure 2, STGCN is composed of several spatio-temporal convolutional blocks, each of which is formed as a "sandwich" structure with two gated sequential convolution layers and one spatial graph convolution layer in between.

3.2 Graph CNNs for Extracting Spatial Features

- In our model, the graph convolution is employed directly on graph structured data to extract highly meaningful patterns and features in the space domain.

- Though the computation of kernel Θ in graph convolution by (2) can be expensive due to $\mathcal{O}(n^2)$ multiplications with graph Fourier basis, two approximation strategies are applied to overcome this issue.

3.2.1 Chebyshev Polynomials Approximation

- To localize the filter and reduce the number of parameters, the kernel Θ can be restricted to a polynomial of Λ as

$$\Theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k,$$

where $\theta \in \mathbb{K}$ is a vector of polynomial coefficients.

- K is the kernel size of graph convolution, which determines the maximum radius of the convolution from central nodes.

- Traditionally, Chebyshev polynomial $T_k(x)$ is used to approximate kernels as a truncated expansion of order $K-1$ as

$$\Theta(\Lambda) \approx \sum_{k=0}^{K-1} \theta_k T_k(\tilde{\Lambda})$$

with rescaled $\tilde{\Lambda} = 2\Lambda/\lambda_{max} - I_n$ (λ_{max} denotes the largest eigenvalue of L)

- The graph convolution can then be rewritten as,

$$\begin{aligned} \Theta *_{\mathcal{G}} x &= \Theta(L)x \\ &\approx \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L})x \end{aligned}$$

, where $T_k(\tilde{L}) \in \mathbb{R}^{n \times n}$ is the Chebyshev polynomial of order k evaluated at the scaled Laplacian $\tilde{L} = 2L/\lambda_{max} - I_n$.