TRR
JOURNAL OF THE TRANSPORTATION RESEARCH BOARD

Research Article

Transportation Research Record
2022, Vol. 2676(12) 409–419
© National Academy of Sciences:
Transportation Research Board 2022
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/03611981221095511
journals.sagepub.com/home/trr
SAGE

Missing Pavement Performance Data Imputation Using Graph Neural Networks

Lu Gao[1], Ke Yu[2], and Pan Lu[3]

Abstract

Pavement condition data is important for providing information on the current state of the network and determining the needs of preventive maintenance or rehabilitation treatments. However, the condition data set is often incomplete for various reasons such as measurement errors and non-periodic inspection intervals. Missing data, especially when missing systematically, presents loss of information, reduces statistical power, and introduces biased assessment. Existing practices in pavement management systems (PMS) usually discard entire cases with missing data or impute it through data correlation. This paper proposes a graph-based deep learning framework, convolutional graph neural networks, to tackle the missing data problem in PMS. Unlike other variants of neural networks, the proposed approach is able to capture the spatio-temporal relationship in data and to learn and reconstruct the missing data by combining information among neighboring sections. In the case study, pavement condition data from 4,446 sections managed by Texas Department of Transportation were used. Experiments show that the proposed model was able to outperform standard machine learning models when imputing the missing data.

Keywords

infrastructure, infrastructure management and system preservation, pavement management systems, pavement management, pavement: asset management

An accurate and complete understanding of pavement asset performance and deterioration processes can not only predict pavement condition over time, but is also essential for optimizing maintenance and rehabilitation programs by allocating budgets more reasonably (*1*). One of the foundations of pavement management systems (PMS) is to predict the condition of the pavement network based on historical condition data (*2*). Although pavement condition is inspected regularly, historical condition data sets are often incomplete for various reasons such as sensor failure or non-periodic inspection (*3*). Missing data are common yet can be problematic in the implementation of predictive modeling.

The subject of missing pavement performance values has been heavily studied. There are two general approaches to address missing data in pavement management: (i) delete the data point with missing values; or (ii) impute or fill in the missing values with techniques such as interpolation, regression, and time-series substitution. The most common imputation strategy is to infer the missing values from the known part of the predictors and data. For example, Al-Zou'bi et al. (*3*) evaluated the effectiveness of various statistical methods (including linear interpolation, linear regression, moving average, cubic regression) in estimating missing values in pavement condition data sets. Farhan and Fwa (*4*) developed an imputation strategy that uses selected pavement properties and nonpavement data as auxiliary variables for missing pavement performance data. The authors tested the proposed strategy on the Long-Term Pavement Performance (LTPP) database and found the proposed approach a useful tool for imputing missing data in pavement management. Karlaftis and Badr (*5*) modeled the probability

[1]University of Houston, Houston, TX
[2]University of Pittsburgh, Pittsburgh, PA
[3]North Dakota State University, Fargo, ND

Corresponding Author:
Lu Gao, lgao5@central.uh.edu

of initiation of alligator cracking following pavement treatment through neural networks using the LTPP database. They used interpolation to treat missing data and assumed that the pavement section cracked at the midpoint of the missing observations. Ziari et al. (*6*) applied neural networks to predict flexible pavement roughness using the LTPP database. In the data pre-processing stage, the authors eliminated observations containing missing data in two continuous years and the interpolation method was used to determine other missing data. Hafez et al. (*7*) evaluated the possibility of collecting pavement condition data less frequently for Wyoming county paved roads. They proposed to estimate the missing pavement condition data using statistical techniques. It was found that the proposed methodology can provide a good estimation of the missing pavement condition indices based on the initial/historical values for the county roads.

Models were also developed to estimate missing road work records. For example, Gao et al. (*8*) proposed a method to estimate the deterioration rate in performance models with missing maintenance records. The method is able to provide the probability of maintenance intervention for each observation based on a Bayesian-based algorithm that can distinguish maintenance or no maintenance. Saliminejad and Gharaibeh (*9*) used Bayesian and spatial statistics to estimate the missing construction and maintenance and rehabilitation (M&R) history of a pavement network from the spatio-temporal patterns of condition data.

Recently, deep learning models have been used for imputation of missing data in the transportation field and were found to outperform standard machine learning models. For example, Duan et al. (*10*) proposed a deep learning model for traffic data imputation. The authors showed that the mean absolute error of the proposed model has better performance compared with other models including the ARIMA model and neural network model. Zhuang et al. (*11*) proposed a convolutional neural network based model to impute missing traffic data. In the proposed model, the raw data is transformed into spatial–temporal images and then applies a deep learning approach to the images. It was found that the proposed model increases the imputation accuracy. Li et al. (*12*) proposed a multimodal deep learning model for traffic data imputation. The model uses two parallel stacked autoencoders, which simultaneously consider the spatial and temporal dependencies. It was found that the proposed model can accurately impute missing data. McMahon et al. (*13*) identified the types of missing data in railway asset management and developed a long short-term memory (LSTM) model (*14*) for imputation of missing data. It was found that the LSTM model is suitable for scenarios where missing data is not completely random and has a strong time-series dependency. Gao et al. (*15*) used deep learning networks to detect if seal coat treatment was applied to a pavement section during a given time period and an accuracy of 87.5% was obtained.

This paper proposes a deep learning approach using the convolutional graph neural network (ConvGNN) for imputing missing data in pavement management. This approach is able to capture spatial and temporal properties simultaneously in the imputation analysis of missing pavement performance data utilizing information from both the node features and the structure of network. The intrinsic features of pavement systems make missing data imputation challenging because the pavement condition data are not only multi-dimensional but also spatial–temporal dependent in a complex way, which has not been addressed by existing studies. The proposed model is evaluated with a case study using pavement management system data from the Texas Department of Transportation. The performance of the proposed approach is also compared with standard machine learning models.

Methodology

The convolutional neural network (CNN) was first proposed by LeCun et al. (*16*). Compared with traditional neural networks, CNN utilizes localized filters among features which allow us to exploit spatial context through neighborhood information for large items. Although there is a direct analogy between image and graph, the traditional convolution operation of CNN cannot be directly applied to graph data because of its irregular format (*17*, *18*). Bruna et al. (*19*) introduced the spectral definition of convolution and generalize CNN on graphs as spectral-based ConvGNN. This allows the extension of CNN to irregular graphs by transforming signals from graph spatial domain to graph spectral domain and applying convolutions as multiplications there.

ConvGNN models have been successfully applied to address many graph-based problems in the transportation area. For example, Wang et al. (*20*) developed a graph convolutional recurrent neural network framework to estimate and predict the spatio-temporal patterns of transportation resilience under extreme weather events. Yu et al. (*21*) proposed a spatio-temporal graph convolutional network to handle a time-series traffic prediction problem. The results conclude that the proposed approach outperforms other models on different traffic datasets. In the following sections, the proposed methodology is discussed in detail.

Pavement Infrastructure Network and Condition Assessment

We begin by considering a general missing data imputation model. Let $\boldsymbol{s}_t \in \mathbb{R}^N$ be the condition assessment of a pavement network of N sections at time t.

$$s_t = f(s_{t-1}, s_{t-2}, ..., s_{t-H}, x, \mathcal{G}) \tag{1}$$

where f is the imputation function which predicts the missing pavement condition in the current time period given the previous H observations. $\boldsymbol{x}$ is a vector of explanatory variables such as traffic, age, pavement type, and maintenance activities. The pavement network is defined as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N nodes $v_i \in \mathcal{V}$ representing sections, edges $(v_i, v_j) \in \mathcal{E}$ indicating the connection between sections. The proposed ConvGNN framework integrates pavement condition inventory data, work history data, and network structure data together. In this way, the model will be able to capture both spatial (information from neighboring sections) and temporal (information from previous time periods) features in imputing the missing values. The spatial distribution of the pavement network is incorporated into the modeling process through its adjacency matrix $\boldsymbol{A}$ and degree matrix $\boldsymbol{D}$ which will be discussed in detail in the next section.

Convolutions on Graphs

The notion of spectral-based ConvGNN is introduced as below. We have the adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N}$, degree matrix $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{A}_{ij}$, and the normalized version of the Laplacian matrix $\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, where $\boldsymbol{\Lambda}$ is a diagonal matrix of $\boldsymbol{L}$'s eigenvalues and $\boldsymbol{U}$ is the matrix of eigenvectors. Given a graph signal $\boldsymbol{x} \in \mathbb{R}^{N \times 1}$ (e.g., pavement condition from previous year), the graph fourier transform is defined as $\tilde{\boldsymbol{x}} = \mathcal{F}(\boldsymbol{x}) = \boldsymbol{U}^T\boldsymbol{x}$ and the inverse transform is $\boldsymbol{x} = \mathcal{F}^{-1} = \boldsymbol{U}\tilde{\boldsymbol{x}}$. Once the signal $\boldsymbol{x}$ is transformed to the spectral domain, a spectral convolution of this signal can be defined using a filter $\boldsymbol{\Theta} = \text{diag}(\vartheta_1, \vartheta_2, ..., \vartheta_N)$. The spectral convolution on graphs is defined as

$$\boldsymbol{\Theta} * \boldsymbol{x} = \mathcal{F}^{-1}\left((\vartheta_1, \vartheta_2, ..., \vartheta_N)^T \odot \mathcal{F}(\boldsymbol{x})\right) = \boldsymbol{U}\boldsymbol{\Theta}\boldsymbol{U}^T\boldsymbol{x} \tag{2}$$

Defferrard et al. (*22*) proposed the following equation to localize the filter and reduce the number of parameters of the model $\boldsymbol{\Theta} = \sum_{j=0}^{K} \theta_j T_j(\boldsymbol{\Lambda})$, where the kernel $\boldsymbol{\Theta}$ is expressed as a Chebyshev polynomial of $\boldsymbol{\Lambda}$ and $\theta_0, \theta_1, ..., \theta_K$ are polynomial coefficients. $T(\cdot)$ is the Chebyshev polynomials and can be recursively expressed as $T_k(\boldsymbol{\Lambda}) = 2\boldsymbol{\Lambda}T_{k-1}(\boldsymbol{\Lambda}) - T_{k-2}(\boldsymbol{\Lambda})$, $T_1(\boldsymbol{\Lambda}) = \boldsymbol{\Lambda}$, and $T_0(\boldsymbol{\Lambda}) = \boldsymbol{I}$. This approach is a kth-order polynomial of the Laplacian matrix and the convolution covers the neighboring nodes K steps away from the central node. Hammond et al. (*23*) proposed rescaling $\boldsymbol{\Lambda}$ as $\hat{\boldsymbol{\Lambda}} = \frac{2\boldsymbol{\Lambda}}{\lambda_{\max}} - \boldsymbol{I}$, where $\lambda_{\max}$ is the largest eigenvalue of $\boldsymbol{L}$. Therefore, the convolution of $\boldsymbol{x}$ becomes

$$\boldsymbol{\Theta} * \boldsymbol{x} \approx \boldsymbol{U}\left(\sum_{j=0}^{K} \theta_j T_j(\hat{\boldsymbol{\Lambda}})\right)\boldsymbol{U}^T\boldsymbol{x} = \sum_{j=0}^{K} \theta_j T_j(\hat{\boldsymbol{L}})\boldsymbol{x} \tag{3}$$

where $\hat{\boldsymbol{L}} = \frac{2\boldsymbol{L}}{\lambda_{\max}} - \boldsymbol{I}$ is the scaled Laplacian matrix. We can generalize the above equation to a signal $\boldsymbol{X} \in \mathbb{R}^{N \times C}$ with C-dimensional feature vector for every node and F filters as

$$\boldsymbol{\Theta} * \boldsymbol{X} \approx \sum_{j=0}^{K} T_j(\hat{\boldsymbol{L}})\boldsymbol{X}\boldsymbol{W}_k \tag{4}$$

where $\boldsymbol{W}_k \in \mathbb{R}^{C \times F}$ is the kth matrix of filter parameters. The convolutional layer defined by Equation 4 is usually called a Chebyshev spectral convolution neural network (ChebConv) (*24*).

Kipf and Welling (*25*) proposed a simplified version of Equation 3. Let $K = 1$, the graph convolution of $\boldsymbol{x}$ becomes

$$\boldsymbol{\Theta} * \boldsymbol{x} \approx (\theta_0\boldsymbol{I} + \theta_1\hat{\boldsymbol{L}})\boldsymbol{x} \tag{5}$$

Furthermore, by assuming that $\lambda_{\max} = 2$ and $\theta = \theta_0 = -\theta_1$, Equation 5 can be further simplified as

$$\boldsymbol{\Theta} * \boldsymbol{x} \approx \theta(\boldsymbol{I} + \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}})\boldsymbol{x} \tag{6}$$

$$= \theta(\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}})\boldsymbol{x} \tag{7}$$

where $\tilde{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{I}$ and $\tilde{\boldsymbol{D}}_{i,i} = \sum_j \tilde{\boldsymbol{A}}_{i,j}$. Equation 6 can be generalized to scenarios where a signal $\boldsymbol{X} \in \mathbb{R}^{N \times C}$ has C -dimensional feature vector for every node. F filters or feature maps are used. The graph convolution of $\boldsymbol{X}$ can be expressed as

$$\boldsymbol{\Theta} * \boldsymbol{X} \approx \tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{X}\boldsymbol{W} \tag{8}$$

where $\boldsymbol{W} \in \mathbb{R}^{C \times F}$ is a matrix of filter/weight parameters. The convolutional layer defined by Equation 8 is usually called graph convolutional network (GCN) (*24*). An intuitive explanation of the above equation is to take the weighted average of all neighbors' node features. When stacking layers together, the number of layers can be considered as the maximum number of hops that the information of each node can travel.

Case Study

Data Description

In this case study, the Pavement Management Information System (PMIS) database (*26*) and work history records collected by the Houston district of Texas Department of Transportation (TxDOT) is used to demonstrate the proposed ConvGNN model. In this model, the current and historical condition data and work records from neighboring sections are used for the prediction of the missing values. A total of 4,446 asphalt pavement sections were selected from the road network managed by the Houston district. Each section is

Table 1. Variables in the Dataset

Variables	Note
Pavement condition indicators	Condition data were collected from 2012 to 2018. Detailed information about indicators can be found in Table 2.
Time since last treatment	The age of a pavement section is calculated as the time difference in years between the last treatment and 2018.
Traffic	The current 18-kip ESAL value for the data collection section. Values are stored in thousands.
Road work records	Treatment types recorded by Texas Department of Transportation. Detailed information about treatment types can be found in Table 3.

Note: ESAL = equivalent single-axle loads.

uniquely identified by columns of ROUTE_NAME, OFFSET_FROM, and OFFSET_TO in the PMIS database. A graph is constructed with its nodes representing the sections and its edges representing the connections between sections. Two sections are considered adjacent to each other if they share the same ROUTE_NAME and one section's OFFSET_FROM is the same as the other section's OFFSET_TO. The section length, functional class, traffic, and number of lanes of these sections are shown in Figure 1. The majority of the sections were taken from farm-to-market roads (FM, BF, FS), interstate highway (IH), state highway (SH, BS, SL, SS), and U.S. highway (US, BU, UA) routes. The rest of the sections belong to park roads (PR). The 20-year cumulative ESALs (equivalent single-axle loads) of these sections range from less than 10,000 to over 60 million. More than 90% of the sections have thick pavement (code 4, 5, and 9). Over 80% of sections have length around 0.5 mi. The dataset used in this study contains key attributes of pavement condition observations from the Texas road network as shown in Table 1.

The key attributes include 12 flexible pavement condition indicators (e.g., rutting, cracking, patching, and roughness) as shown in Table 2. Among these 12 indicators, three them represent the general condition of a road pavement: Distress Score, Ride Score, and Condition Score. Distress Score reflects the amount of visible surface deterioration of a pavement. It ranges from 1 (the most distress) to 100 (the least distress). Ride Score is a measure of the pavement's roughness. It ranges from 0.1 (the roughest) to 5.0 (the smoothest). Condition Score represents the pavement's overall condition in relation to both distress and ride quality. It ranges from 1 (the worst condition) to 100 (the best condition). Other indicators include shallow rutting, deep rutting, patching, failures, block cracking, alligator cracking, longitudinal cracking, transverse cracking, and international roughness index (IRI). Figure 2 shows the distribution of these 12 condition indicators in the year 2018.

In this case study, pavement maintenance and rehabilitation history data collected from TxDOT Houston district were also used. The maintenance dataset contains information about the type of treatments, when they were implemented, and which pavement sections they were applied to. The type of treatments are shown in Table 3. The work history records were converted to dummy variables representing each individual treatment when used in the modeling process.

Experimental Settings

Figure 3 shows the architecture of the proposed ConvGNN model that is composed of two graph convolutional layers, each of which is activated using the Rectified Linear Unit (ReLU) function (*27*, *28*), and one dense layer. The ReLU function is defined as the positive part of its input: $f(x) = \max(0, x)$.

Both ChebConv (*4*) and GCN (*8*) layers of ConvGNN model were tested in the case study. With the above three-layer architecture, the form of the forward model with the ChebConv layers is

$$\hat{\boldsymbol{s}}_t = \mathrm{ReLU}\left[\sum_{j^{(2)}=0}^{K^{(2)}} T_{j^{(2)}}(\hat{\boldsymbol{L}})\left(\mathrm{ReLU}\left[\sum_{j^{(1)}=0}^{K^{(1)}} T_{j^{(1)}}(\hat{\boldsymbol{L}})\boldsymbol{X}\boldsymbol{W}_k^{(1)}\right]\right)\boldsymbol{W}_k^{(2)}\right]\boldsymbol{W}^{(3)} \tag{9}$$

where $\hat{\boldsymbol{s}}_t$ represents the estimated complete condition assessment of the pavement network at time period t. $\boldsymbol{X}$ represents the feature matrix including historical condition assessment, maintenance and rehabilitation work records, and traffic information. The superscript represents the index of the layer. For example, $\boldsymbol{W}^{(3)}$ denotes the weight matrix of the third layer (dense layer) in Figure 3. The form of the model with the GCN layers is

$$\hat{\boldsymbol{s}}_t = \mathrm{ReLU}\left[\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\left(\mathrm{ReLU}\left[\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{X}\boldsymbol{W}^{(1)}\right]\right)\boldsymbol{W}^{(2)}\right]\boldsymbol{W}^{(3)} \tag{10}$$

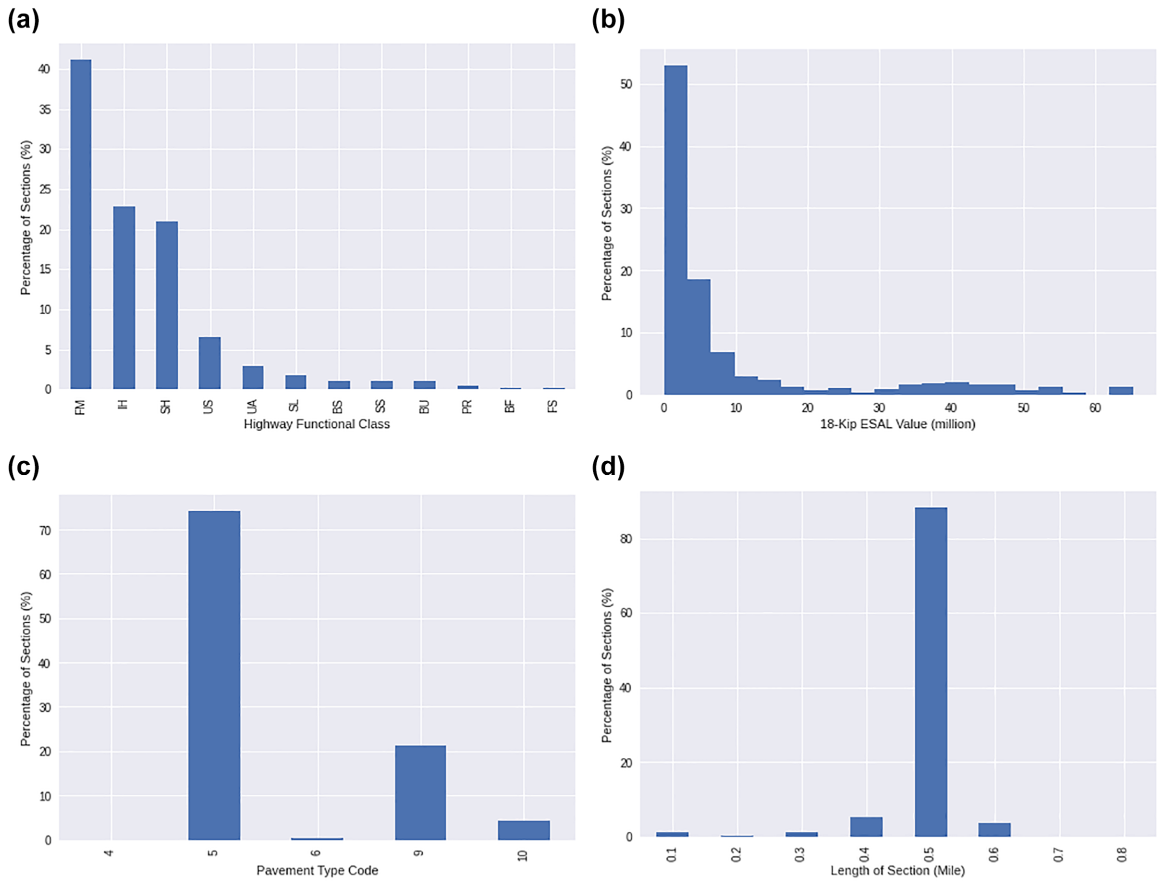

Figure 1. Information on sections used in the case study: (*a*) functional class, (*b*) 18 kip equivalent single-axle loads (ESALs), (*c*) pavement type, and (*d*) length of section.

Note: Code indicating pavement types: 4 = thick asphalt concrete (greater than 5.5 in.), 5 = medium thickness asphalt concrete (2.5–5.5 in.), 6 = thin asphalt concrete (less than 2.5 in.), 9 = overlaid and widened asphalt concrete pavement, 10 = thin surfaced flexible pavement (surface treatment or seal coat).

Table 2. Flexible Pavement Surface Condition Indicators

#	Condition indicator	Unit	Range
1	Shallow rutting (0.25–0.49 in. depth)	Percentage	0–100
2	Deep rutting (0.50–0.99 in. depth)	Percentage	0–100
3	Patching	Percentage	0–100
4	Failures	Quantity	⩾ 0
5	Block cracking	Percentage	0–100
6	Alligator cracking	Percentage	0–100
7	Longitudinal cracking	Foot	⩾ 0
8	Transverse cracking	Quantity	⩾ 0
9	International roughness index (IRI)	Inch/mile	⩾ 0
10	Ride score	na	0–5
11	Distress score	na	0–100
12	Condition score	na	0–100

Note: na = not applicable.

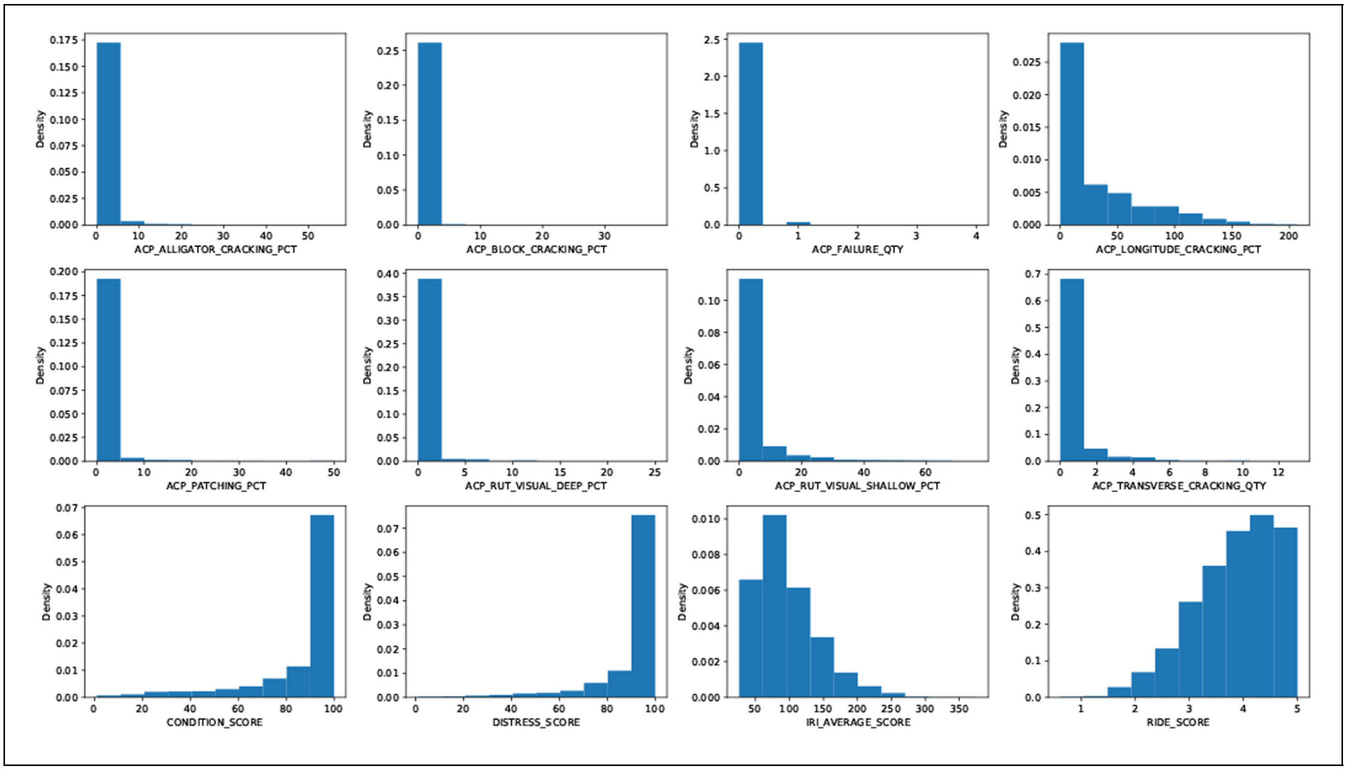

Figure 2. Histograms of condition indicators in 2018.

Table 3. Road Work Type

#	Work description
1	SC—Seal coat
2	RER—Rehabilitation of existing road
3	OV—Overlay
4	P05—Full width seal coat
5	RES—Restoration
6	RMS—Routine maintenance project (sealed)

Further a mean squared error loss function (Equation 11) was used for the optimization process and the sections with missing data are assigned with values minimizing the loss function. Using the features data, $\boldsymbol{X}$, and the target data, $\boldsymbol{s}_t$, back-propagation can be conducted through forward models (Equation 9) or (Equation 10) to optimize the weight matrix, $\boldsymbol{W}$, with respect to the loss function (Equation 11).

$$\mathcal{L} = \frac{1}{N}(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t)^T(\boldsymbol{s}_t - \hat{\boldsymbol{s}}_t) \tag{11}$$

The proposed methodology was used to build 12 models. For each model, the target variable is the 2018 data of one condition indicator from Table 2 and the features are the rest of the data in Table 1. For example, when 2018 IRI data is used as the target variable, all 12 condition indicators' observations for 2012–2017 (previous six years) are used together with pavement age, traffic, and treatment type as features. By assuming that part of the target data records is missing, we can evaluate the performance of these models in predicting the missing condition values. The inclusion of the historical condition data from 2012 to 2017 enables the models to capture the non-linearity in the deterioration process of condition indicators.

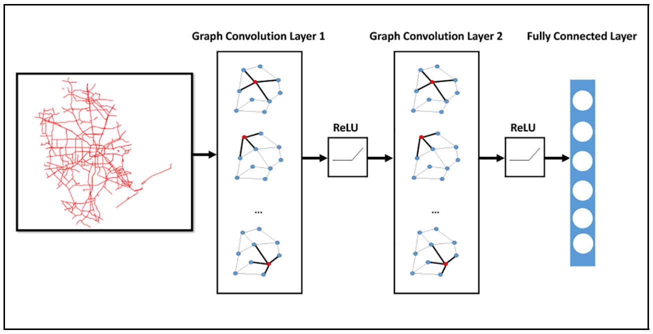

Figure 3. Architecture of the convolutional graph neural network (ConvGNN) model.

The following procedure is used to test the performance of the proposed model in predicting the missing data. The study used 20% of the data for testing and 80% for training. The dataset was split in two different

Table 4. Performance Comparison of Different Models on Missing Data in Clusters

Indicator	CART	LR	NN	RF	ConvGNN
Alligator cracking	−0.327	0.150	0.273	0.376	0.441
Block cracking	−0.775	0.071	0.040	0.229	0.252
Failure	−0.559	0.057	0.084	0.111	0.163
Longitudinal cracking	−0.212	0.163	0.233	0.435	0.476
Patching	−0.416	0.122	0.216	0.387	0.452
Deep rutting	0.018	−1.05	0.478	0.544	0.591
Shallow rutting	0.246	−6.72	0.606	0.649	0.720
Transverse cracking	−0.244	0.176	0.081	0.394	0.450
Condition score	−0.006	0.446	0.500	0.521	0.570
Distress score	−0.130	0.363	0.326	0.453	0.529
IRI	0.685	0.844	0.842	0.861	0.886
Ride score	0.720	0.852	0.853	0.866	0.894

Note: CART = classification and regression trees; LR = linear regression; NN = neural network; RF = random forest; ConvGNN = convolutional graph neural network; IRI = international roughness index.

approaches. In the first approach, pavement sections are randomly selected across the network to be placed in the testing dataset. This represents the scenario where the missing data occurs by chance. In the second approach, the test sections are randomly distributed in clusters where the condition data of their neighboring sections are also missing. This represents the situation where part of the network is not inspected. The reason for setting up those two scenarios is twofold: (i) it represents two important missing data patterns; and (ii) since the ConvGNN model passes messages to neighboring sections, it is necessary to test how the model works in different neighborhood settings. To implement the second scenario in the case study, first a route is randomly selected and consecutive sections from that route used in the testing dataset. Similar results were obtained from both approaches.

A Boolean mask was set for each of the sections in both training and testing datasets. The masks are *true* when they belong to the corresponding dataset. The graph convolutional networks model is trained through a semi-supervised approach, where the whole graph is used for both training and testing. The masks are implemented as sample weights when training the model. For each road section in the testing dataset, the actual condition indicator value in 2018 is labeled as missing, and the condition indexes in the previous years and other variables are used as features to predict the missing target values. The model is trained using the whole graph as the input. Each node of the graph is considered as a pavement section. The training set consists of a subset of graph nodes on which the target indicator is evaluated and gradients are calculated through back-propagation. As a semi-supervised regression model, the graph sections with missing indicator values are also part of the training process and their features affect the convolution layers. All experiments are compiled and tested on a Windows desktop (CPU: Intel octa-core i7-3940XM CPU @ 3.00GHz, 32GB RAM). The ConvGNN models were implemented in PyTorch Geometric (PyG), which is a Python library supporting many types of deep learning on graphs. PyG makes it easy to build a deep learning model though customizing predefined graph neural network layers (*29*).

The proposed model is also compared with the following machine learning models as baselines: (i) classification and regression trees; (ii) neural network; (iii) linear regression; and (iv) random forest. R2-score (Equation 12) is used to measure and evaluate the performance of different models.

$$\text{R2} - \text{score} = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y}^i)^2} \tag{12}$$

where $\hat{y}_i$ is the predicted value of the i th data point, y_i is the actual value of the i th data point, and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$. The best possible value for r2-score is 1. A model that predicts y_i to be a constant value $\bar{y}$ will result in a r2-score of 0. The R2-score can also be negative when the model performs worse than predicting everything to be $\bar{y}$.

Results

Two different layers of ConvGNN were tested: the Chebyshev and the GCN. The former gives the best results when the Chebyshev polynomial order $K = 2$. The results of the proposed model and baselines are given in Table 4, which lists the r2-score for each condition indicator. The proposed ConvGNN model, which integrates information from the immediate neighboring road segments and their neighborhood, provides the best performance, achieving better performance in all of the

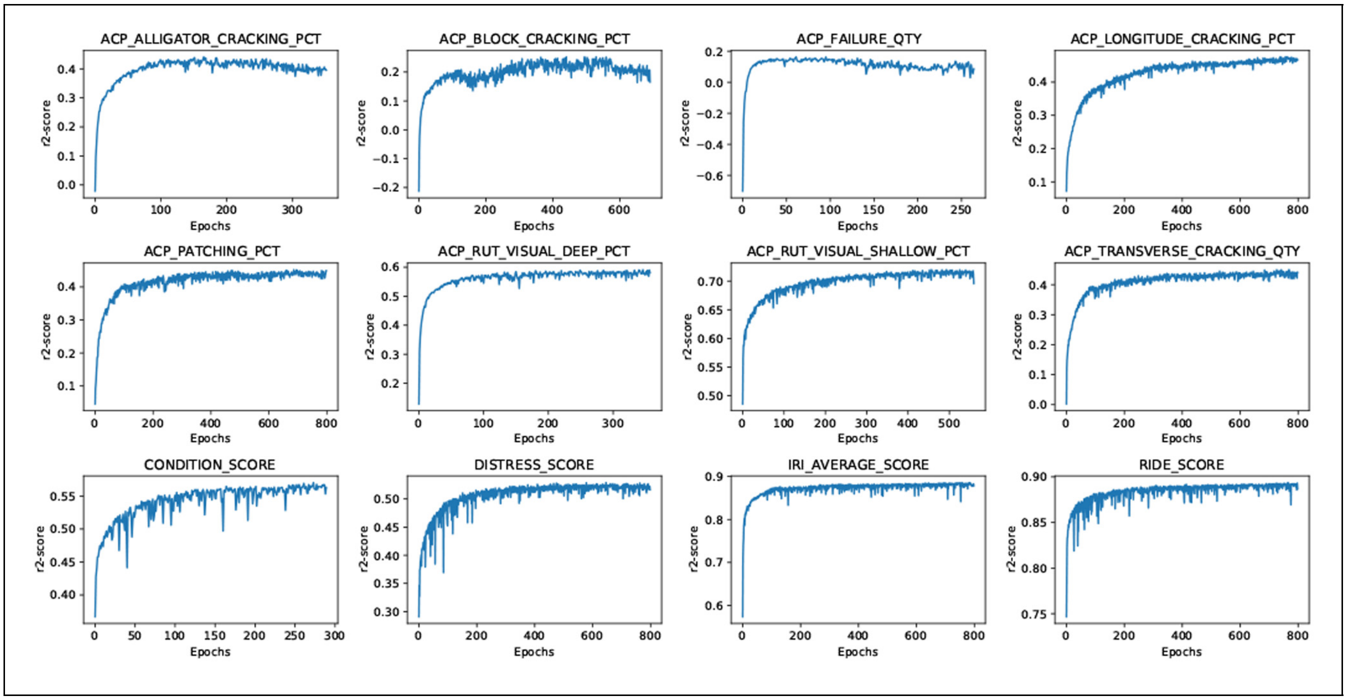

Figure 4. Testing dataset r2-score versus epoch of convolutional graph neural network (ConvGNN) model.

12 condition indicators than machine learning regression models using the Scikit-Learn library (*30*). The random forest model has the best performance among machine learning models. The worst performance is observed for the decision tree with most of its r2-scores being negative. The best performances observed are for the IRI (0.886) and Ride Score (0.894), both of which are measurements of the pavement roughness and Ride Score is just a linear transformation of IRI. The reasons why roughness models have better results than other indicators are probably twofold: (i) the IRI measurement data are found to be more consistent among adjacent sections than other distress indicators; and (ii) the deterioration of roughness (i.e., changes between consecutive years) is found to be more linear compared with other distress indicators. The estimation result for shallow rutting results also has r2-score above 0.7, which represents that 70% of the variance in the target variable can be explained by the model. The r2-scores for other condition indicators are between 0.16 and 0.59. The comparison between the actual values and the predictions on the testing dataset is shown in Figure 5. Based on the results from the case study, the proposed approach produces reasonably good results in predicting missing data for roughness and shallow rutting. While the random forest model gives the best results among all machine learning models, the deep learning graph neural network model is able to improve the results by another 5% in r2-score on average. The r2-score curves of the testing datasets are shown in Figure 4. The modeling results are able to converge and achieve the highest r2-score after 300 to 800 epochs.

Conclusion

This paper proposes a deep learning framework for predicting missing condition data for pavement networks through graph convolutional layers. The results show that the model outperforms other machine learning models on a real-world dataset. The pavement network was modeled as a graph combining historical condition data and spatial connections between neighboring segments. This graph is used to train ConvGNN models in a semi-supervised approach to learn on a subset of nodes with known condition data and evaluation on the rest where data is missing. The case study shows that the proposed approach is expected to produce good results in imputing missing roughness and shallow rutting data. Particularly, the indicators related to estimation of roughness achieve high r2-scores close to 0.9. While the proposed approach outperforms the traditional machine learning models, the difference between the deep learning model and the random forest model is within 5% on average. Since building and executing a deep learning model takes much longer than a random forest model, it is recommended that highway agencies use the latter when computational resources are not available.

The developed model can assist engineers and administrators in more effectively managing infrastructure

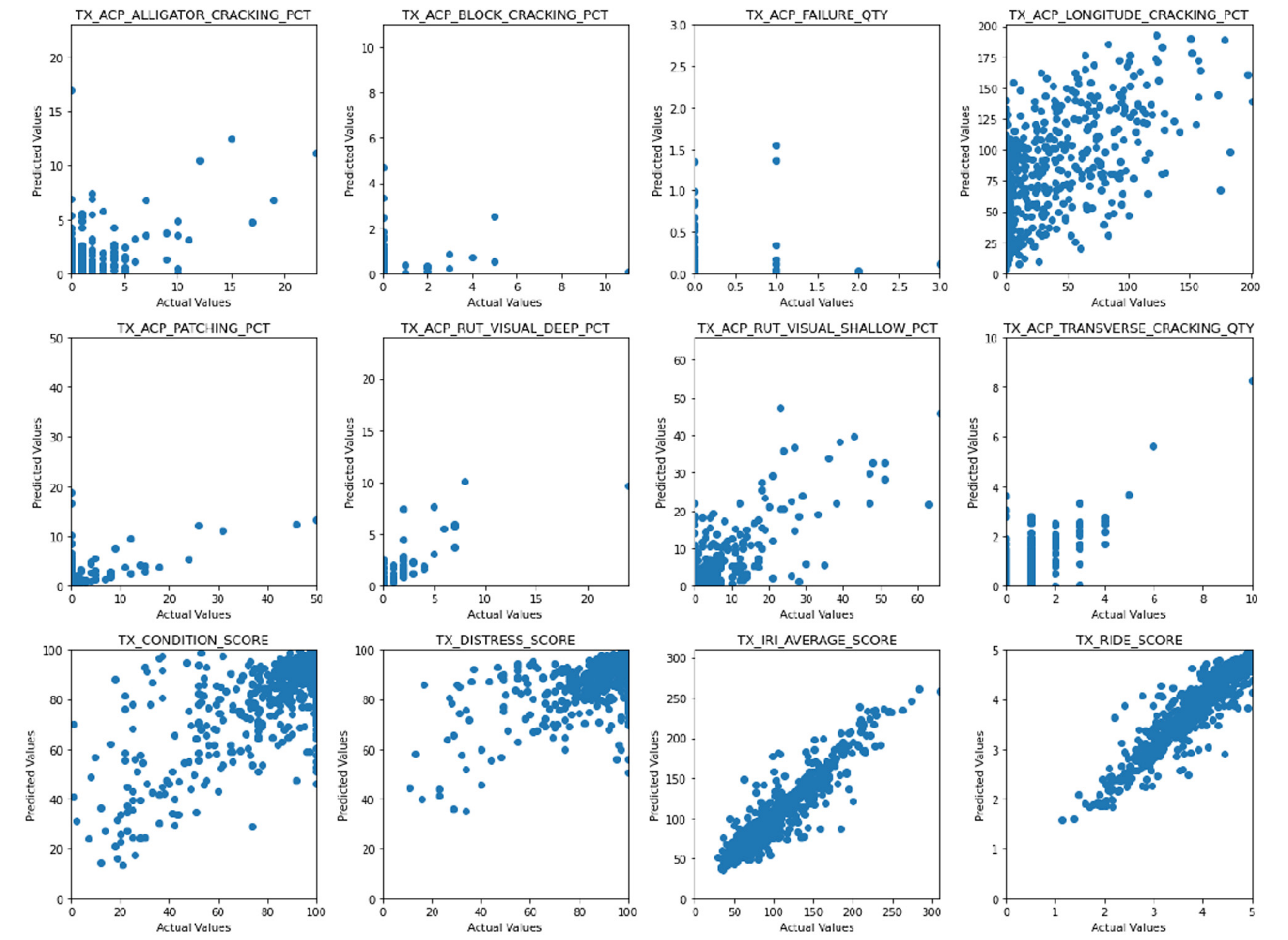

Figure 5. Actual values versus predicted values.

systems through improved performance prediction; in particular, it can help pavement data quality management at the network level. Once trained with historical data, the proposed model can also be used for predicting future pavement conditions. However, caution should be taken when preparing the testing datasets in verifying the trained model. For the problem of missing data prediction, the training and testing datasets can be collected from the same year. To build a model predicting future conditions, the training and testing datasets should be collected from different time periods so that the prediction power of the trained model can be verified. Furthermore, although the case study in this paper was developed and tested for pavement deterioration, it can be easily implemented and extended to characterize the performance of other infrastructure facilities. More accurate prediction of missing condition data will better assist decision makers to select maintenance and rehabilitation treatments.

Acknowledgments

The authors would like to thank all Texas Department of Transportation personnel, who have helped this research study.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: L. Gao, K. Yu; data collection: L. Gao; analysis and interpretation of results: L. Gao, P. Lu; draft manuscript preparation: L. Gao, K. Yu, P. Lu. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Texas Department of Transportation Grant 0-6988.

ORCID iDs

Lu Gao https://orcid.org/0000-0003-2421-2000
Ke Yu https://orcid.org/0000-0001-9882-5729
Pan Lu https://orcid.org/0000-0002-1640-3598

References

1. Li, Z. *A Probabilistic and Adaptive Approach to Modeling Performance of Pavement Infrastructure*. PhD thesis. The University of Texas, Austin, 2005.
2. Haas, R., and W. R. Hudson. *Pavement Management Systems*. McGraw-Hill, New York, NY, 1978.
3. Al-Zou'bi, M. M., C. M. Chang, S. Nazarian, and V. Kreinovich. Systematic Statistical Approach to Populate Missing Performance Data in Pavement Management Systems. *Journal of Infrastructure Systems*, 21, No. 4, 2015, p. 04015002.
4. Farhan, J., and T. F. Fwa. Improved Imputation of Missing Pavement Performance Data Using Auxiliary Variables. *Journal of Transportation Engineering*, Vol. 141, No. 1, 2015, p. 04014065.
5. Karlaftis, A. G., and A. Badr. Predicting Asphalt Pavement Crack Initiation Following Rehabilitation Treatments. *Transportation Research Part C: Emerging Technologies*, Vol. 55, 2015, pp. 510–517.
6. Ziari, H., J. Sobhani, J. Ayoubinejad, and T. Hartmann. Prediction of IRI in Short and Long Terms for Flexible Pavements: ANN and GMDH Methods. *International Journal of Pavement Engineering*, Vol. 17, No. 9, 2016, pp. 776–788.
7. Hafez, M., K. Ksaibati, and R. Anderson-Sprecher. Utilizing Statistical Techniques in Estimating Uncollected Pavement-Condition Data. *Journal of Transportation Engineering*, Vol. 142, No. 12, 2016, p. 04016065.
8. Gao, L., J. P. Aguiar-Moya, and Z. Zhang. Performance Modeling of Infrastructure Condition Data With Maintenance Intervention. *Transportation Research Record: Journal of the Transportation Research Board*, 2011. 2225: 109–116.
9. Saliminejad, S., and N. G. Gharaibeh. A Spatial-Bayesian Technique for Imputing Pavement Network Repair Data. *Computer-Aided Civil and Infrastructure Engineering*, Vol. 27, No. 8, 2012, pp. 594–607.
10. Duan, Y., Y. Lv, Y. -L. Liu, and F. -Y. Wang. An Efficient Realization of Deep Learning for Traffic Data Imputation. *Transportation Research Part C: Emerging Technologies*, Vol. 72, 2016, pp. 168–181.
11. Zhuang, Y., R. Ke, and Y. Wang. Innovative Method for Traffic Data Imputation Based on Convolutional Neural Network. *IET Intelligent Transport Systems*, Vol. 13, No. 4, 2018, pp. 605–613.
12. Li, L., B. Du, Y. Wang, L. Qin, and H. Tan. Estimation of Missing Values in Heterogeneous Traffic Data: Application of Multimodal Deep Learning Model. *Knowledge-Based Systems*, Vol. 194, 2020, p. 105592.
13. McMahon, P., T. Zhang, and R. A. Dwight. Approaches to Dealing With Missing Data in Railway Asset Management. *IEEE Access*, Vol. 8, 2020, pp. 48177–48194.
14. Hochreiter, S., and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, Vol. 9, No. 8, 1997, pp. 1735–1780.
15. Gao, L., Y. Yu, Y. H. Ren, and P. Lu. Detection of Pavement Maintenance Treatments Using Deep-Learning Network. *Transportation Research Record: Journal of the Transportation Research Board*, 2021. 2675: 1434–1443.
16. LeCun, Y., B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel. Handwritten Digit Recognition with a Back-Propagation Network. In *Advances in Neural Information Processing Systems* (M. I. Jordan, Y. Lecun, and A. A. Solla, eds.), MIT Press, Cambridge, MA, 1990, pp. 396–404.
17. Niepert, M., M. Ahmed, and K. Kutzkov. Learning Convolutional Neural Networks for Graphs. *Proc., International Conference on Machine Learning, PMLR*, 2016, pp. 2014–2023.
18. Battaglia, P. W., J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, et al. Relational Inductive Biases, Deep Learning, and Graph Networks. *arXiv Preprint* arXiv:1806.01261, 2018.
19. Bruna, J., W. Zaremba, A. Szlam, and Y. LeCun. Spectral Networks and Locally Connected Networks on Graphs. *arXiv Preprint* arXiv:1312.6203, 2013.
20. Wang, H. -W., Z. -R. Peng, D. Wang, Y. Meng, T. Wu, W. Sun, and Q. -C. Lu. Evaluation and Prediction of Transportation Resilience Under Extreme Weather Events: A Diffusion Graph Convolutional Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 115, 2020, p. 102619.
21. Yu, B., H. Yin, and Z. Zhu. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. *arXiv Preprint* arXiv:1709.04875, 2017.
22. Defferrard, M., X. Bresson, and P. Vandergheynst. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *Proc., 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 3844–3852.
23. Hammond, D. K., P. Vandergheynst, and R. Gribonval. Wavelets on Graphs via Spectral Graph Theory. *Applied and Computational Harmonic Analysis*, Vol. 30, No. 2, 2011, pp. 129–150.
24. Wu, Z., S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32, No. 1, 2020, pp. 4–24.
25. Kipf, T. N., and M. Welling. Semi-Supervised Classification With Graph Convolutional Networks. *arXiv Preprint* arXiv:1609.02907, 2016.
26. Gharaibeh, N., T. Freeman, S. Saliminejad, A. Wimsatt, C. Chang-Albitres, S. Nazarian, I. Abdallah, et al. *Evaluation and Development of Pavement Scores, Performance Models and Needs Estimates for the TxDOT Pavement Management Information System*. Technical Report. Texas Transportation Institute, College Station, 2012.

27. Nair, V., and G. E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. *Proc., 27th International Conference on Machine Learning*, Omnipress, Haifa, Israel, 2010.
28. Xu, B., N. Wang, T. Chen, and M. Li. Empirical Evaluation of Rectified Activations in Convolutional Network. *arXiv Preprint* arXiv:1505.00853, 2015.
29. Fey, M., and J. E. Lenssen. Fast Graph Representation Learning With PyTorch Geometric. *Proc., ICLR Workshop on Representation Learning on Graphs and Manifolds*, International Conference on Learning Representations (ICLR), 2019.
30. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. Scikit-Learn: Machine Learning in Python. *The Journal of Machine Learning Research*, Vol. 12, 2011, pp. 2825–2830.

All opinions, errors, omissions, and recommendations in this paper are the responsibility of the authors and are not those of the Texas Department of Transportation. This paper does not constitute a standard, specification, or regulation.