Short-Term Traffic Prediction Based on DeepCluster in Large-Scale Road Networks

Lingyi Han, *Student Member, IEEE*, Kan Zheng, *Senior Member, IEEE*, Long Zhao, *Member, IEEE*, Xianbin Wang, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Short-term traffic prediction (STTP) is one of the most critical capabilities in Intelligent Transportation Systems (ITS), which can be used to support driving decisions, alleviate traffic congestion and improve transportation efficiency. However, STTP of large-scale road networks remains challenging due to the difficulties of effectively modeling the diverse traffic patterns by high-dimensional time series. Therefore, this paper proposes a framework that involves a deep clustering method for STTP in large-scale road networks. The deep clustering method is employed to supervise the representation learning in a visualized way from the large unlabeled dataset. More specifically, to fully exploit the traffic periodicity, the raw series is first divided into a number of sub-series for triplet generation. The convolutional neural networks (CNNs) with triplet loss are utilized to extract the features of shape by transforming the series into visual images. The shape-based representations are then used to cluster road segments into groups. Thereafter, a model sharing strategy is further proposed to build recurrent NNs-based predictions through group-based models (GBMs). GBM is built for a type of traffic patterns, instead of one road segment exclusively or all road segments uniformly. Our framework can not only significantly reduce the number of prediction models, but also improve their generalization by virtue of being trained on more diverse examples. Furthermore, the proposed framework over a selected road network in Beijing is evaluated. Experiment results show that the deep clustering method can effectively cluster the road segments and GBM can achieve comparable prediction accuracy against the IBM with less number of prediction models.**

***Index Terms*—Short-term traffic prediction, large-scale road networks, deep representation learning, deep clustering.**

I. Introduction

SHORT-TERM traffic prediction (STTP) techniques have recently been studied for efficient route planning and traffic control in Intelligent Transportation Systems (ITS) [1]–[3]. The main idea of STTP is to predict the road traffic state (i.e., flow, speed and density) in the next five to thirty minutes by analyzing historical traffic data [4]. Existing STTP studies mainly focused on one road segment, or a small-scale road network that contains only a few adjacent road segments. However, effective route planning requires a global perspective based on the information of the whole network [5]–[9]. The most existing STTP methods are limited to a single scenario such as freeway, arterial and corridor. They are difficult to be generalized to a *heterogeneous* road network that involves several road segments of diverse road functions. Besides, the development of ITSs within cities increases the amount of traffic data in terms of time span and granularity [10]. Therefore, making full use of tremendous traffic data to improve accuracy of STTP in large-scale networks becomes a challenge.

The existing STTP method of large-scale networks is either to develop a specific prediction model with the high accuracy for each road segment, termed as individual-based model (IBM), or build a general prediction model with the high efficiency for all road segments, termed as whole-based model (WBM). Considering the multiplicity and heterogeneity of large-scale networks, neither of the two models is appropriate for STTP in large-scale networks due to the following reasons. On the one hand, massive number of IBMs occupy lots of storage resources in ITS. Besides, the number of training samples collected from one segment is insufficient for developing a robust IBM due to the easy trends towards overfitting. On the other hand, WBM is not effective for modeling the whole network with different types of traffic patterns. Moreover, the WBM is vulnerable to the curse of dimensionality by taking historical data from all segments as inputs. Therefore, both efficiency and accuracy of the two models are not ideal for large-scale networks. To overcome this difficulty, a feasible STTP method for large-scale networks needs to be studied.

Generally, representation learning, a.k.a. dimension reduction, is used to transform the raw data into a good representation that makes the subsequent learning easy. It plays an important role in time series clustering, because time series are essentially high-dimensional and susceptible to noise. Hence, clustering of raw series is computationally expensive and distance measures are highly sensitive to the distortions. Recently, deep learning (DL) has witnessed many success in different related areas, including computer vision, speech recognition and networking [11]–[15], due to its theoretical function approximation

Manuscript received June 10, 2019; revised August 21, 2019; accepted October 6, 2019. Date of publication October 14, 2019; date of current version December 17, 2019. This work was supported by the National Natural Science Foundation of China under Grant 61731004. The review of this article was coordinated by Dr. B. Mao. *(Corresponding author: Kan Zheng.)*

L. Han, K. Zheng, and L. Zhao are with the Intelligent Computing and Communication (IC^2) Laboratory, the Wireless Signal Processing and Networks Laboratory (WSPN), and the Key Laboratory of Universal Wireless Communications, Beijing University of Posts and Telecommunications, Ministry of Education, Beijing 100088, China (e-mail: hanlingyi@bupt.edu.cn; zkan@bupt.edu.cn; z-long@bupt.edu.cn).

X. Wang is with the Department of Electrical and Computer Engineering, University of Western Ontario, London, ON N6A 5B9, Canada (e-mail: xianbin.wang@uwo.ca).

X. Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TVT.2019.2947080

0018-9545 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

properties [16], and feature learning capabilities [17]. To leverage these technological advancements, deep representation is adopted in this paper for traffic series clustering.

In this paper, an STTP framework composed of a deep clustering method and several prediction models is proposed for large-scale road networks. The proposed framework aims to achieve a good tradeoff between the quantity of prediction models and the prediction accuracy. The existing DL-based representation learning methods in clustering are mostly designed for static data. They may not suit for the traffic series data, which is higher dimensional and more sensitive to noise than static data. Instead of directly applying the existing DL-based representation learning methods, a novel shape-based representation learning method is developed for road segments clustering. The deep clustering method is self-driven and is capable of processing a large number of high-dimensional traffic series data. After clustering, by exploiting the diversity and similarity of traffic patterns, a group-base model (GBM) is proposed for a type of traffic patterns. The main technical contributions of the paper are summarized as follows:

- A traffic series clustering method based on DL representation learning termed as DeepCluster, is developed. By fully exploiting the periodicity of traffic data, a strategy is designed to generate triplets[1] from original unlabeled traffic series. A triplet consists of three sub-series that two of them have more similar profiles than others, which are then used in turn to supervise the representation learning. Moreover, the dimension of the series used for representation learning is significantly reduced as compared to raw series.
- Unlike the existing hand-craft features, such as the frequency transformation, wavelet transformation and Shapelets, a pure data-driven representation learning method is proposed to learn the shape-based features of traffic series. More specifically, a rasterization strategy is designed to transform the traffic series into traffic images in a visualized way. The convolutional neural network (CNN) with triplet loss is used for traffic series representation learning. At last, the representations are used to cluster the road network into K groups by clustering methods.
- Instead of modeling a road segment or all road segments, the GBM is proposed for a type of traffic patterns. Based on the idea of model sharing, all road segments in a group with similar traffic profiles share one prediction model. Each GBM is learned by aggregating all the training samples from a group. Model sharing increases the number and the diversity of the training samples, which is beneficial for model generalization. Experiment results validate that the GBM has stronger generalization ability than IBM. The impact of input interval on prediction performance is also analyzed by experiments.

The rest of paper is organized as follows. Section II reviews the related works. In Section III, the data used throughout the paper is described. Section IV formulates the STTP problem in large-scale networks and introduces the proposed framework. In Section V, the DeepCluster method are detailed. The DeepPrediction is then proposed in Section VI. In Section VII, simulation results of the proposed framework are given, before concluding the paper in Section VIII.

II. Related Works

A. Time Series Representation Learning

A wide variety of methods had been developed for time series representation learning in clustering [18]–[20], such as spectral transformation [21], wavelets transformation [21], eigenvalue analysis techniques [22], piecewise linear approximation (PLA) [23], adaptive piecewise constant approximation (APCA) [24], symbolic approximation (SAX) [25], piecewise aggregate approximation (PAA) [26] and perceptually important point (PIP) [27]. However, all these methods are based on hand-craft features, which are designed to describe specific time series patterns and heavily rely on the database.

A new trend of time series learning merges with artificial neural networks (ANNs), especially DNNs based representation learning in clustering, which are data-driven and capable of learning a powerful representation from raw data through a high-level and non-linear mapping. Therefore, some works have used the deep representation learning to improve the performance of clustering. C. Song *et al.* in [28] integrated K-means algorithm into a stacked auto-encoder (SAE) by minimizing the reconstruction error as well as the distance between data points and corresponding clusters. It alternatively learned the representations and updated cluster centers. In [29], [30], the K-means algorithm used the nonlinear representations that are learned by DNNs for clustering. J. Xie *et al.* in [31] proposed a deep embedded clustering method that simultaneously learned the representations and cluster assignments. It defined a centroid-based probability distribution and minimizing its Kullback-Leibler (KL) divergence to an auxiliary target distribution. K. Tian *et al.* in [32] improved the existing works by proposing a general flexible framework that integrated traditional clustering methods into different DNNs. The framework is optimized by alternating direction of multiplier method (ADMM). However, the above methods all worked with the static data that are simple and low dimensional compared with time series data in general. On the other hand, there is less research on the deep representation learning of time series in clustering. Therefore, an efficient time series representation learning algorithm dedicated for clustering needs to be developed.

B. Short Term Traffic Prediction

There are numerous researches on single-point STTP [5], such as autoregressive integrated moving average (ARIMA) family of models, Kalman filters, support vector machine (SVM), traffic flow theory-based simulation models, ANNs and recurrent NNs (RNNs). Obviously, single-point models predict the future traffic state for a target road segment only using its own historical data, which ignores the relations between the target road segment and adjacent segments. Consequently, some researches have focused on predicting one or multiple segments by taking the

[1] A triplet consists of two traffic series with the same positive tag and a traffic series with a negative tag.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

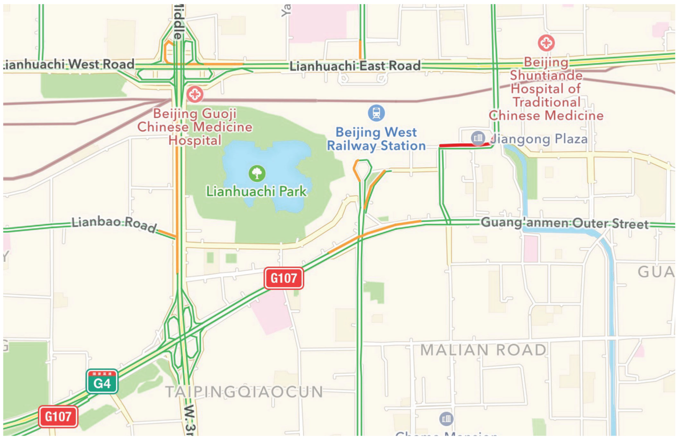


Fig. 1. The network topology at Liuli Bridge, Beijing.

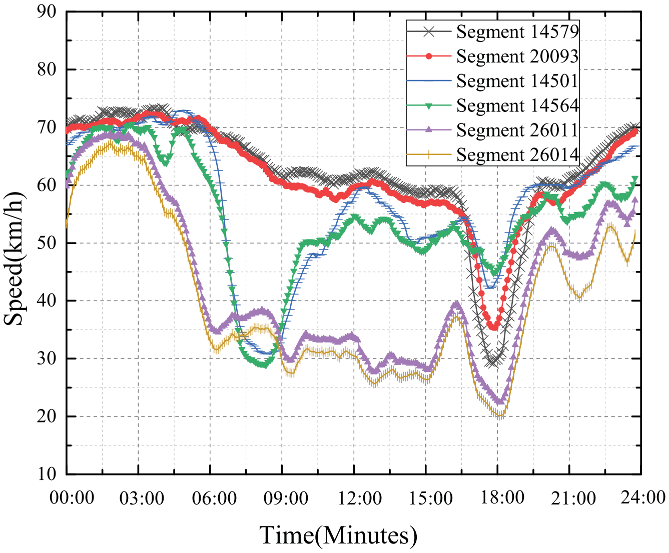


Fig. 2. Average traffic speeds of six road segments with five-minute time interval from September, 2017 to November, 2017.

spatio-temporal interrelations among adjacent road segments into account [33]–[37]. However, the above network-level STTP methods belong to IBMs or WBMs, which restricted to small regions that containing several adjacent road segments.

Recently, a few literatures begin to pay attention to the predictions in large-scale networks. In [38], dynamic simulator based on traffic flow theory was used for STTP in the network with limited traffic data. [39]–[41] only predicted the traffic state of the representative road subset to achieve the prediction for a whole network by utilizing data compression technologies. However, the performance of prediction was poor resulted from compression and reconstruction errors. W. Min *et al.* in [42] considered a road network consists of about 500 road segments. However, they developed a custom model for the test area, which is not practical. Z. Zhao *et al.* in [43] performed prediction for each individual road segment with a two-dimensional long short term memory (LSTM) network by considering spatio-temporal correlations. A comparison with other representative forecast models including ARIMA, SVM, basic RNN, etc. validates that the proposed LSTM network can achieve a better performance. Nevertheless, STTP through IBMs is hard to implement for large-scale networks in practice. in [44], [45], DNNs such as deep belief networks (DBNs) and CNNs are employed to improve the performance of STTP. However, they only built one model and expected it to fit for all segments without considering the fact that the whole network is heterogeneous with different types of segments. Therefore, these attempts are hard to be implemented in large-scale networks with high accuracy.

III. The Data

The traffic data used throughout the paper is collected from Beijing Liuli Bridge, the road network topology of which is shown in Fig. 1. The network consists of about 1,000 road segments with a diverse level of road functions including express way, arterial road, access road, side road, etc. The dataset collected by Beijing Transportation Institute contains the traffic speeds from September, 2017 to November, 2017 with five-minute sampling interval. Hence, each road segment has totally 90 × 288 measured data, where 90 means the number of days and 288 means the number of speed samples collected in each day. The speed data is measured by vehicles such as taxis and buses that are equipped with GPS.

IV. Proposed Framework for STTP

A. Formulation of STTP Problem

Consider a large-scale road network Φ consists of N_{road} road segments, i.e., $\Phi = \{\boldsymbol{x}^{(r)}\}_{r=1}^{N_{\text{road}}}$, where $\boldsymbol{x}^{(r)} = [x_1^{(r)}, x_2^{(r)}, \ldots, x_{N_{\text{speed}}}^{(r)}]$ represents the average speed series on the rth segment. A sub-series of $\boldsymbol{x}^{(r)}$ is denoted by

$$\boldsymbol{x}^{(r)}_{n:L:l} = [x^{(r)}_n, x^{(r)}_{n+l}, \ldots, x^{(r)}_{n+(L-1)l}], \tag{1}$$

where $\boldsymbol{x}^{(r)}_{n:L:l}$ is a set of L continuous measured values with sampling intervals l from a time series $\boldsymbol{x}^{(r)}$, that starts at position n. $\boldsymbol{x}^{(r)}_{n:L:1}$ is abbreviated as $\boldsymbol{x}^{(r)}_{n:L}$ for simplicity.

Let $\hat{x}_{n+N_{\text{output}}}$ be the predicted speed value with prediction horizon N_{output}, given the corresponding N_{input} historical speed values up to time n. The goal of STTP is to construct a mapping function $f(\cdot)$ between the historical speed values and the future one, i.e.,

$$\begin{aligned}\hat{x}_{n+N_{\text{output}}} &= f(x_{n-N_{\text{input}}+1}, x_{n-N_{\text{input}}+2}, \ldots, x_n)\\ &= f(\boldsymbol{x}_{n-N_{\text{input}}+1:N_{\text{input}}}).\end{aligned} \tag{2}$$

B. Proposed Framework

As stated above, neither IBMs nor WBMs is suitable for large-scale networks, because they consist of not only a large number of road segments, but also a variety of road segment types, as shown in Fig. 2. To tackle this problem, a framework dedicated for STTP in large-scale road networks is proposed in this paper. The proposed framework clusters the road segments into several groups, each of which has a typical traffic pattern. Within each

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

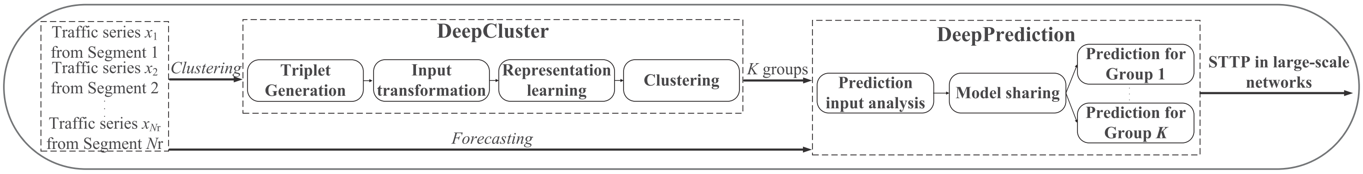


Fig. 3. The block diagram of the proposed STTP framework.

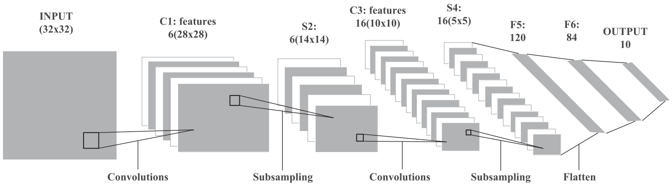


Fig. 4. Architecture of LeNet-5 (Above the rectangles are the number of channels and its size in parenthesis).

group, the traffic patterns of all road segments are highly similar. Based on that, an STTP model is built for a type of the traffic patterns within a group rather than a road segment or all road segments.

The architecture of this framework is illustrated in Fig. 3. It consists of two major components, i.e., DeepCluster and DeepPrediction. The DeepCluster method is proposed for representation learning and clustering of high-dimensional traffic series from a unlabeled traffic dataset. Based on the clustering results, the DeepPrediction method is developed for effective STTP. The DeepCluster and DeepPrediction are detailedly given in Section V and VI, respectively.

V. DeepCluster

In this section, the proposed DeepCluster method is described in details. Before presenting the DeepCluster, the problem of clustering in large-scale networks is formulated as follows.

Definition 1: Given a large-scale network Φ consists of N_{road} traffic series, i.e., $\Phi = \{\boldsymbol{x}^{(r)}\}_{r=1}^{N_{\text{road}}}$, the process of partitioning Φ into K groups $\{\boldsymbol{C}^{(1)}, \boldsymbol{C}^{(2)}, \ldots, \boldsymbol{C}^{(K)}\}$ is called *traffic series clustering*. In such a way that homogenous traffic series are grouped together based on a certain similarity measure.

In contrast to the traditional extrinsic handcrafted features, human can easily seize the intrinsic visual-based features, which is why they can quickly distinguish different types of the time series under the help of high abstraction ability. Moreover, compared with raw time series, the intrinsic visual-based features are much steadier and can be less affected by the distortions and scale of samples. To overcome the issues of raw data-based or handcraft-based clustering methods, deep representation learning is used for traffic series clustering.

The key advantage of CNNs is that the features are not designed by handcrafting, but are learned from data using a general-purpose learning procedure [17]. CNNs can process any form of arrays, such as 1D series, 2D images and 3D videos. Fig. 4 shows the architecture of a typical CNN, named LeNet-5 [48], where two main types of layers different with the regular ANNs are convolutional layers (C-layer in Fig. 4) and subsampling layers (S-layer in Fig. 4). The convolutional and subsampling operators make the representations more invariant to the distortion compared to the raw data. Besides, the parameter sharing makes the CNNs capable of processing high-dimensional inputs. The aforementioned characteristics inspire us to adopt the CNNs for traffic series representation learning. In this section, we explore an efficient deep CNN architecture, i.e., FaceNet [46], to learn a mapping from raw high-dimensional traffic series to low-dimensional representations that are used for clustering.

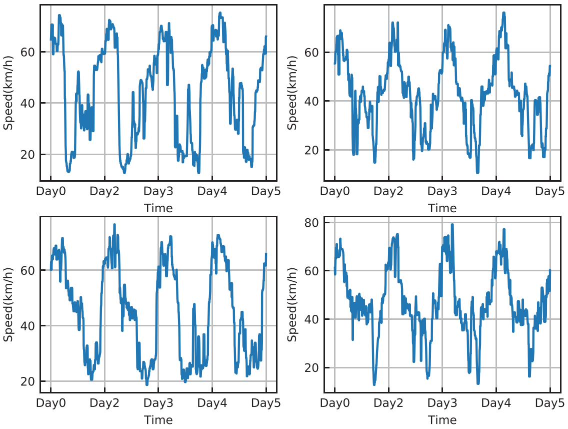


Fig. 5. Average traffic speeds of four road segments on different days of the week from September, 2017 to November, 2017. (Five-minute time interval)

The DeepCluster includes triplet generation, inputs transformation, representation learning and clustering, which is given below in details.

A. Triplet Generation

As can be seen in Fig. 5, the daily traffic patterns mostly follow the same trend. The traffic daily similarity index defined in [47] is calculated to investigate the traffic periodic pattern of one day. The traffic daily similarity is defined as the normalized gaps between each pair of measurements in two consecutive days from one road segment. As stated in Section III, the traffic speeds are collected every 5 minutes. Since one day has 288 time intervals, the traffic similarity $\text{SIM}_m^{(r)}$ in Segment r in time interval m can be calculated by

$$\text{SIM}_m^{(r)} = \frac{|x_m^{(r)} - x_{m+288}^{(r)}|}{\max_{1\le n\le N_{\text{speed}}-288} |x_n^{(r)} - x_{n+288}^{(r)}|}. \tag{3}$$

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

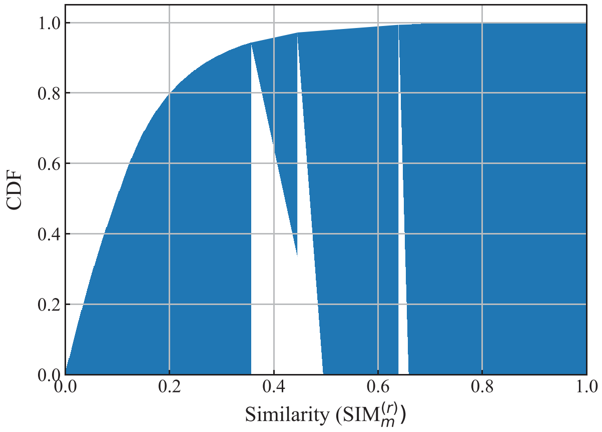


Fig. 6. The CDF of similarity ($\mathrm{SIM}_m^{(r)}$) with real traffic speeds. The low $\mathrm{SIM}_m^{(r)}$ near 0 indicates that the speed is almost the same with the speed at the same daily time from consecutive day.

The small value indicates that the speed changes little with respect to the speed at the same daily time from consecutive day. Fig. 6 demonstrates the cumulative distribution function (CDF) of $\mathrm{SIM}_m^{(r)}$ over all road segments. It can be clearly observed that more than 80% $\mathrm{SIM}_n^{(r)}$ are smaller than 0.2, which indicates that periodic pattern exists in traffic series on most of the road segments. To fully exploit the traffic temporal features of periodicity, we propose to generate triplets consist of three traffic series that two of them are more similar than others in profile. Given N_{road} traffic series with period N_{period} measured from N_{road} road segments, the traffic series is split into sub-series by periods, termed as periodic sub-series. Thus, $d = N_{\text{speed}}/N_{\text{period}}$ periodic sub-series for each segment are obtained,

$$\{\boldsymbol{x}^{(r)}_{1:N_{\text{period}}}, \boldsymbol{x}^{(r)}_{N_{\text{period}}+1:N_{\text{period}}}, \ldots, \boldsymbol{x}^{(r)}_{(d-1)N_{\text{period}}+1:N_{\text{period}}}\}, \tag{4}$$

here $\boldsymbol{x}^{(r)}_{nN_{\text{period}}+1:N_{\text{period}}} = [x^{(r)}_{nN_{\text{period}}+1}, \ldots, x^{(r)}_{(n+1)N_{\text{period}}}]$ is the $(n+1)$th ($0 \leq n \leq d-1$) periodic sub-series in Segment r. According to the traffic daily similarity, a triplet is made up by randomly choosing two different periodic sub-series from one Segment r_i, and one sub-series from another Segment r_j,

$$\{\boldsymbol{x}^{(r_i)}_{nN_{\text{period}}+1:N_{\text{period}}}, \boldsymbol{x}^{(r_i)}_{mN_{\text{period}}+1:N_{\text{period}}}, \boldsymbol{x}^{(r_j)}_{kN_{\text{period}}+1:N_{\text{period}}}\},$$
$$0 \leq n, m, k \leq d-1, n \neq m, r_i \neq r_j. \tag{5}$$

In a triplet, two sub-series from a segment are marked as positive tags, while the other is marked as negative tag. The triplets are then used to supervise the traffic series representation learning. The employment of triplets can be regarded as a strong prior belief imposed on the representations of traffic series. The prior belief indicates that the traffic sub-series sampled from a road segment should be clustered into a group in most cases.

B. Inputs Transformation

In order to extract the shape-based features, a rasterization strategy is designed to visualize the traffic series into gray-scale images, as shown in Fig. 7. The transformed images can reveal the features of shape well, such as bulge and sink. Let the series $\boldsymbol{x} = [x_1, x_2, \ldots, x_N]$ be standardized by min-max normalization to keep values between 0 and 1. The N dimensional series $\boldsymbol{x}$ is transformed to a $N \times N$ dimensional matrix $\boldsymbol{X}$ with each element being expanded to a N dimensional vectors,

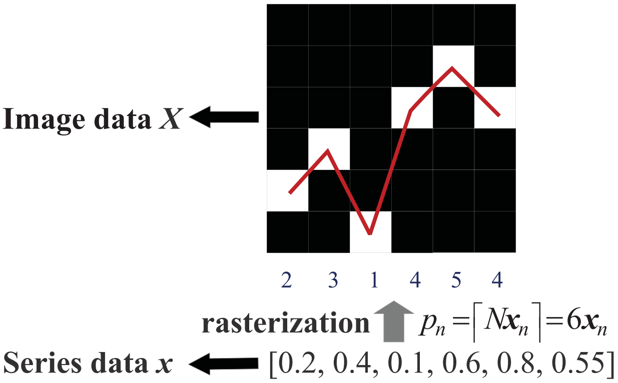


Fig. 7. The schematic diagram of the inputs transformation.

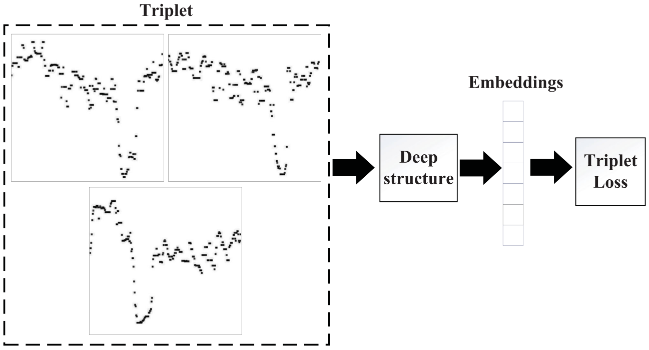


Fig. 8. The block diagram of representation learning.

$$\boldsymbol{X} = [\boldsymbol{e}^{255}(p_1), \boldsymbol{e}^{255}(p_2), \ldots, \boldsymbol{e}^{255}(p_N)], \tag{6}$$

where $\boldsymbol{e}^{255}(p_n)$ is a N dimensional column vector corresponding to the nth element x_n of $\boldsymbol{x}$. $\boldsymbol{e}^{255}(p_n)$ denotes a vector with the pixel value of 255 at its p_nth element standing for white and 0 standing for black elsewhere. The position p_n can be calculated by

$$p_n = \lceil N x_n \rceil, \quad p_n \in \{1, 2, \ldots, N\}, \tag{7}$$

where $\lceil \cdot \rceil$ denotes the operator of rounding up. The transformed images are shown in Fig. 8, which are used as inputs for representation learning. The sub-image corresponding to the sub-series $\boldsymbol{x}^{(r)}_{nN_{\text{period}}+1:N_{\text{period}}}$ is denoted as $\boldsymbol{X}^{(r)}_n$. Therefore, the triplet becomes

$$\{\boldsymbol{X}^{(r_i)}_n, \boldsymbol{X}^{(r_i)}_m, \boldsymbol{X}^{(r_j)}_k\},$$
$$0 \leq n, m, k \leq d-1, n \neq m, r_i \neq r_j. \tag{8}$$

C. Representation Learning and Clustering

DNNs with triplet loss from [46] is employed to learn the representations of traffic data from a traffic image space into a feature space. The triplet loss encourages the representations of

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

a pair of sub-images from one segment to be close to each other, and those from different segments to be far away in the feature space. Denoting $f(\boldsymbol{X})$ as the representation of $\boldsymbol{X}$, the triplet loss is given by

$$\|f(\boldsymbol{X}_n^{(r_i)}) - f(\boldsymbol{X}_m^{(r_i)})\|_2^2 - \|f(\boldsymbol{X}_n^{(r_i)}) - f(\boldsymbol{X}_k^{(r_j)})\|_2^2, \quad \forall\{\boldsymbol{X}_n^{(r_i)}, \boldsymbol{X}_m^{(r_i)}, \boldsymbol{X}_k^{(r_j)}\} \in \Gamma, \tag{9}$$

where Γ is the set of all possible triplets. The structure of DNNs with triplet loss is shown in Fig. 8, where the outputs of the last layer are the representations used for clustering. The representations are not only learned by the current sub-image, but also other sub-images coming from a road segment. The dimension of the representations is lower than the raw series. For example, considering a traffic series with five-minute interval during 90 days. The length of whole series is 288 × 90, while the length of daily sub-series is 288. If 32-dimensional representations are used in clustering, the ratio of reduction in dimension is about $\frac{288\times 90-32}{288\times 90} \approx 99\%$. Subsequently, all the representations from one road segment are averaged, and the averaged representations are then clustered into k groups, i.e.,

$$\boldsymbol{C}^{(k)} = \{\boldsymbol{x}^{(r,k)}\},\ 1 \le k \le K, 1 \le r \le N_{\text{road}}, \tag{10}$$

where K is much less than N_{road}. $\boldsymbol{C}^{(k)}$ denotes the kth group, and $\boldsymbol{x}^{(r,k)}$ represents that the rth road segment in the network clustered into the kth group $\boldsymbol{C}^{(k)}$.

VI. DeepPrediction

Based on the clustering results, a DeepPrediction module including of prediction input analysis and model sharing is further developed for STTP in large-scale networks in this section. Some definitions and statements are given first.

Definition 2: Given two functions $g^{(1)} : \mathbb{R} \to \mathbb{R}$ and $g^{(2)} : \mathbb{R} \to \mathbb{R}$, $g^{(1)}$ is *homogeneous* with $g^{(2)}$ if there exists a real number α such that

$$g^{(1)}(x) = g^{(2)}(x + \alpha),\ \forall x \in R. \tag{11}$$

Theorem 1: Let $\{(x_i, y_i^{(1)})\}_{i=1}^N$ be N distinct successive samples generated from a function $g^{(1)}$. Given a prediction model $f(\cdot; \boldsymbol{w})$ with parameters $\boldsymbol{w}$, build a map between historical values and the future value: $y_N^{(1)} = f(y_1^{(1)}, y_2^{(1)}, \ldots, y_{N-1}^{(1)}; \boldsymbol{w}^{(1)})$. Similarly, get N successive samples $\{(x_i + \alpha, y_i^{(2)})\}_{i=1}^N$ from another function $g^{(2)}$ and build a map $y_N^{(2)} = f(y_1^{(2)}, y_2^{(2)}, \ldots, y_{N-1}^{(2)}; \boldsymbol{w}^{(2)})$ with the same prediction model. Then $\boldsymbol{w}^{(1)}$ is equal to $\boldsymbol{w}^{(2)}$ if $g^{(1)}$ is *homogeneous* with $g^{(2)}$ at step α.

As described in Eq. 2, the prediction model only utilizes the historical speed values, and *Theorem* 1 indicates that if two speed curves are *homogeneous*, the prediction models are identical. A prediction model that specific for a type of traffic patterns instead of a road segment or all segments is proposed. Unlike the regular ANNs, RNNs are capable of exhibiting the temporal correlations of time series, which makes them applicable to tasks such as language modeling, speech recognition and time series forecasting. The key idea of RNNs is to imitate a sequential dynamic behavior with a chain-like structure that allows the information to be passed from previous layer to the current one, as illustrated in Fig. 9. In this paper, RNNs are used as prediction models to capture the temporal correlations of traffic series. The implementations of the DeepPrediction are elaborated as follows.

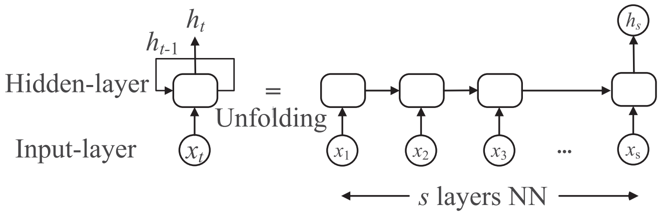


Fig. 9. Architecture of a basic three-layer RNN.

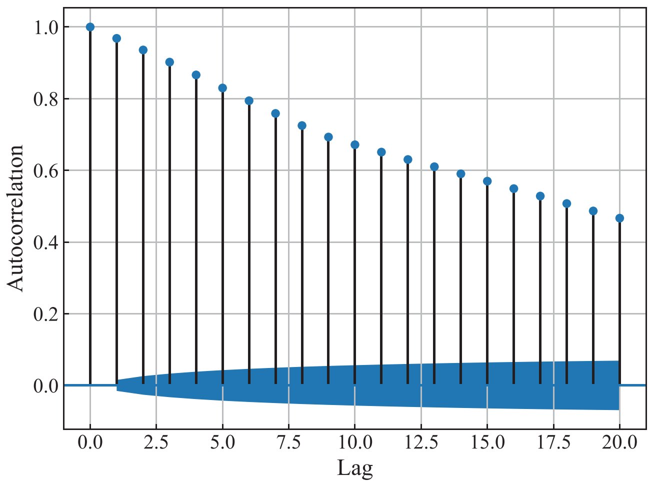


Fig. 10. The ACF of the traffic speeds of a random segment from lag 1 to lag 20. The shaded area represents the 95% confidence intervals, which is used to determine whether the autocorrelation coefficients is significantly different from zero.

A. Prediction Input Analysis

The parameters of inputs of prediction models, including the time span and interval are analyzed in this section. Based on the observations that the traffic data has the similar daily patterns in Section V-A, the time span of prediction inputs is set to be a day, to provide the enough historical information, as well as avoid the information redundancy between days. In order to further reduce redundancy within days, we investigate the autocorrelation of traffic series. The autocorrelation function (ACF) at lag i represents the correlations between measurements which are i intervals apart. As shown in Fig. 10, the traffic speeds on the consecutive intervals are linearly correlated, which implies that feeding all measurements of a day to the prediction model may result in information redundancy. The input interval l used in the prediction model is calculated by

$$l = \max_{i \ge 1}\{i : p_i > p\}, \tag{12}$$

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

where p_i denotes the ACF at lag i, and p is the given threshold that is determined by experiments. Hence, the input series $\boldsymbol{x}_{n-N_{\text{input}}+1:N_{\text{input}}}$ becomes $\boldsymbol{x}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l}$. The length of the input reduces from $N_{\text{input}} = N_{\text{period}}$ to $N^*_{\text{input}} = \lceil N_{\text{period}}/l \rceil$.

B. Model Sharing

Within each group, a prediction model termed as group-based model (GBM) is trained for all road segments. The training samples for each group are generated by

$$\begin{aligned} &\boldsymbol{x}^{(r,k)} \rightarrow < \boldsymbol{x}^{(r,k)}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l},\ x^{(r,k)}_{n+N_{\text{ouput}}} >, \\ &\boldsymbol{x}^{(r,k)} \in \boldsymbol{C}^{(k)}, \end{aligned} \tag{13}$$

where $\boldsymbol{x}^{(r,k)}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l}$ and $x^{(r,k)}_{n+N_{\text{ouput}}}$ denote the input and output of the prediction model, respectively. After that, the samples within a group are aggravated to train a GBM $f^{(k)}(\cdot)$ for Group $\boldsymbol{C}^{(k)}$,

$$x^{(r,k)}_{n+N_{\text{output}}} = f^{(k)}(\boldsymbol{x}^{(r,k)}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l}),\ \boldsymbol{x}^{(r,k)} \in \boldsymbol{C}^{(k)}. \tag{14}$$

Then, the STTP model $f(\cdot)$ of the road network can be written as

$$\begin{aligned} x^{(r,k)}_{n+N_{\text{output}}} &= f(\boldsymbol{x}^{(r,k)}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l}) \\ &= f^{(k)}(\boldsymbol{x}^{(r,k)}_{n-N_{\text{input}}+1:N^*_{\text{input}}:l}),\ k \in \{1, 2, \ldots, K\}. \end{aligned} \tag{15}$$

The proposed DeepPrediction method has the following advantages:

- As compared to the IBMs, the number of prediction models is significantly reduced.
- The increase of the number of training samples and the decrease of the number of input dimensions can not only improve the generalization of prediction models, but also avoid the curse of dimensionality [49].

VII. Performance Evaluation

In this section, the proposed framework for the road network mentioned in Section III is evaluated. The data used in our experiments are introduced at first. Then, the experiment setup and performance metrics are described. The prediction performances over different metrics are finally analyzed.

A. Experiment Setup

The missing historical data adversely affect the performance of traffic prediction [50]. Therefore, the missing ratio of each road segment is calculated over the road network stated in Section III. The missing ratio represents the percentage of missing data in a road segment during the sampling time span. On the one hand, a prediction model is built for each road segment to make a comparison with IBMs. On the other hand, too much missing data influence the clustering results significantly, since the road segments are clustered into groups by the traffic profiles. By taking both the experimental complexity and data integrity into accounts, the road segments whose missing ratios are lower than 90% are used for performance evaluations. The number of eligible road segments is 27 in total.

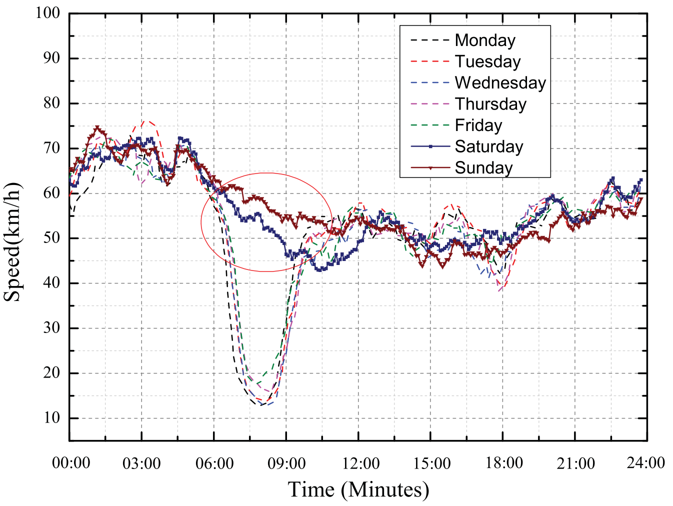
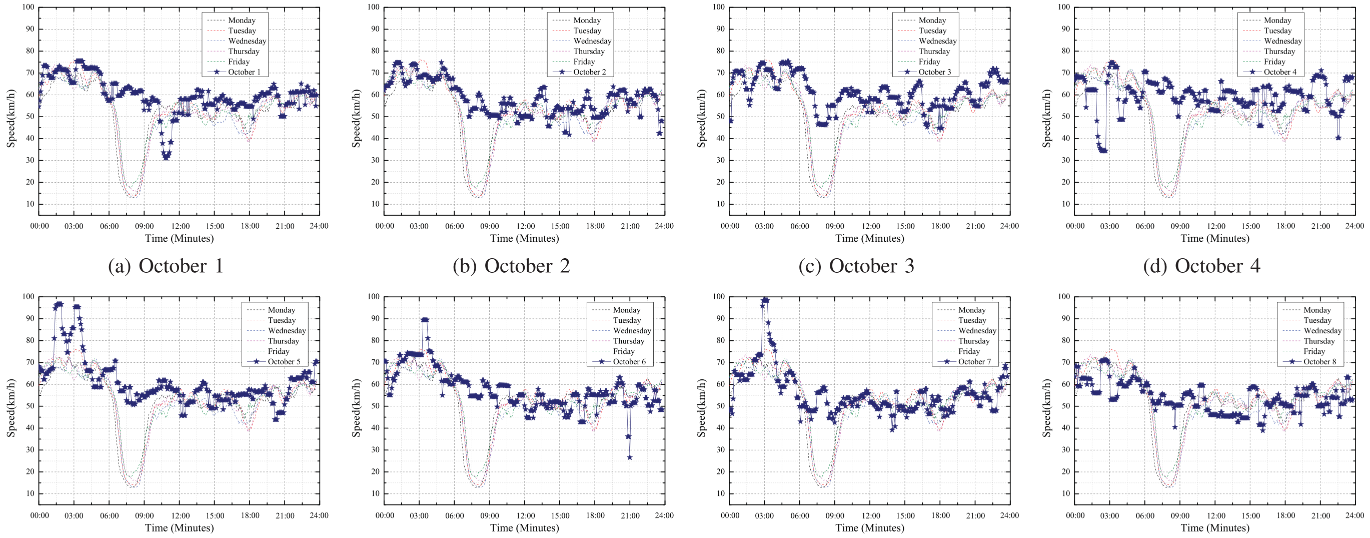


Fig. 11. Average traffic speeds of one random segment on weekdays versus those at weekends with five-minute time interval from September, 2017 to November, 2017.

In DeepCluster module, the traffic series is first split into 90 daily sub-series of length 288 for each segment. Fig. 11 shows that the traffic patterns on weekdays are different from those at weekends between six and ten o'clock in the morning, since most people do not work at weekends (The circular region). Besides, the National Day, i.e., from October 1 to October 8 in 2017, has a great influence on traffic patterns. Fig. 12 shows that the traffic patterns behave abnormally during these days. It is better to learn the representations of traffic patterns of weekdays, weekends and National Day separately and use them for clustering. However, the number of daily sub-series on the weekends and holidays is too small to learn the good representations of the traffic patterns. As for the National Day, there are only eight daily sub-series for a road segment. Even worse, the eight daily sub-series do not follow the similar patterns and we need to learn a representation for each day of the holiday. Therefore, we only consider the workday in the data set and the proposed framework can be directly extended to the cases with weekends or holidays. As a result, 60 daily sub-series are chosen by deleting those at weekends and during the National Day. Then, the sub-series of size 1 × 288 is transferred into images of size 288 × 288. As discussed in Section V, triplets are generated by the daily sub-series from 27 road segments, which are used for representation learning. The deep structure of FaceNet used in this paper is the Inception_ResNet, the configuration of which is the same with that in [46]. The averaged representations of the sub-series from a road segment are then used for clustering by K-means algorithm.

In DeepPrediction module, it is worth mentioning that there is no limitation on the type of time series forecasting algorithms. The state of the art RNN, i.e., LSTM [51] is used for experimentations. As stated in Section VI-A, the time span of inputs is

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.



(a) October 1 (b) October 2 (c) October 3 (d) October 4

(e) October 5 (f) October 6 (g) October 7 (h) October 8

Fig. 12. Average traffic speeds of a random segment on weekdays with five-minute time interval from September, 2017 to November, 2017 versus those during the National Day.

TABLE I
THE CONFIGURATIONS OF THE RELEVANT DNNs

Module	**Network**	**Parameter**	**Size**
DeepCluster	^{a}FaceNet	Image size	160
		Batch size	12
		Segments per batch	6
		Images per segment	9
		Embedding size	32
DeepPrediction	^{b}LSTM	Time steps	$\lceil 288/l \rceil$
		LSTM1	$[1 \times 50]$
		LSTM2	$[50 \times 25]$
		Dense1	$[25 \times 200]$
		Dense2	$[200 \times 1]$

[a] The implications of the parameters given in this table are explained exactly in [46].
[b] The inputs and outputs size are described in $[rows \times cols]$.

chosen to be a day. Then, the number of inputs fed to LSTM is $N_{\text{input}} = \lceil 288/l \rceil$, which is determined by the following experiments. The minimum prediction horizon is determined by the sampling interval. In order to analyze the impacts of prediction horizons on prediction performance, each LSTM is set to have one output node, which predicts the next N_{output}th speed. The training samples belonging to the same group are aggregated to train the LSTM, and then K LSTM GBMs are used for STTP in the road networks.

Key parameters of the relevant DNNs are listed in Table I. If not mentioned specifically, all prediction models are trained by eighty percent of data while tested by the remaining data. 10-fold cross-validation is adopted over training dataset. K-means clustering method is implemented by the Scikit-learn package in Python 3.6.5. The NNs are conducted with a NVIDIA p2000 GPU, TensorFlow r1.8, CUDA 9.0 and CuDNN 9.0. Moreover, four performance metrics including of relative error (RE), mean relative error (MRE), max mean relative error (MARE) and minimum mean relative error (MIRE) are used for evaluation, which are defined as

$$e_{\text{RE}}^{(r)} = \frac{\left| x_{n+N_{\text{output}}}^{(r)} - \hat{x}_{n+N_{\text{output}}}^{(r)} \right|}{x_{n+N_{\text{output}}}^{(r)}}, \; 1 \le r \le N_{\text{road}}, \quad (16)$$

$$e_{\text{MRE}}^{(k)} = \frac{1}{|\boldsymbol{C}^{(k)}|} \sum_{\boldsymbol{x}^{(r,k)} \in \boldsymbol{C}^{(k)}} e_{\text{RE}}^{(r,k)}, \; 1 \le k \le K, \quad (17)$$

$$e_{\text{MARE}}^{(k)} = \max_{\boldsymbol{x}^{(r,k)} \in \boldsymbol{C}^{(k)}} \{e_{\text{RE}}^{(r,k)}\}, \; 1 \le k \le K, \quad (18)$$

$$e_{\text{MIRE}}^{(k)} = \min_{\boldsymbol{x}^{(r,k)} \in \boldsymbol{C}^{(k)}} \{e_{\text{RE}}^{(r,k)}\}, \; 1 \le k \le K, \quad (19)$$

where $e_{\text{RE}}^{(r)}$ denotes the RE of the rth segment with $x_{t+N_{\text{output}}}^{(r)}$ being the true speed and $\hat{x}_{t+N_{\text{output}}}^{(r)}$ being the prediction. Denoting $e_{\text{RE}}^{(r,k)}$ as the RE of the rth segment clustered into group $\boldsymbol{C}^{(k)}$. $|C^{(k)}|$ is the number of road segments in kth group. Besides, $e_{\text{MRE}}^{(k)}$, $e_{\text{MARE}}^{(k)}$ and $e_{\text{MIRE}}^{(k)}$ are MRE, MARE and MIRE of Group k, respectively. The network-level MRE can be similarly calculated by

$$e_{\text{MRE}} = \frac{1}{N_{\text{road}}} \sum_{\boldsymbol{x}^{(r)} \in \Phi} e_{\text{RE}}^{(r)}. \quad (20)$$

B. Road Segments Clustering

In order to better determine the number of clusters, four different clustering criteria are selected, including of Dunn

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

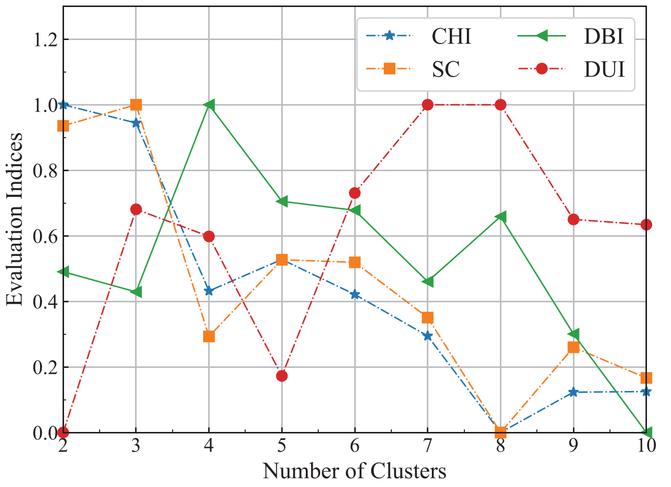


Fig. 13. Comparison of CHI, SC, DBI and DUI with respect to the number of clusters.

TABLE II
THE PERFORMANCE UNDER DIFFERENT INPUT INTERVALS

Input interval l	MRE of Training(%)	MRE of Testing(%)
1	3.70	5.77
3	4.18	5.25
5	4.37	5.48
7	5.83	8.77

Index (DUI) [52], Calinski-Harabasz index (CHI) [53], Davies-Bouldin index (DBI) [54] and Silhouette coefficient (SC) [55]. The higher CHI, SC and DUI indicate the better number of clusters, while the lower DBI indicates the better one. Fig. 13 evaluates each clustering criterion with respect to the number of clusters. It can be clearly found that 3 is the optimal number of clusters by comprehensively considering the four criteria.

All 27 road segments are clustered into 3 groups by DeepCluster, as shown in Fig. 14. It can be found that the traffic series in the same group are in general *homogeneous* with the other series, which demonstrates the proposed DeepCluster's ability of extracting the shape-based features. For example, the traffic speeds in Group 1 have a breakdown in traffic speed during the evening peak period, followed by speed recovery. The traffic speeds in Group 2 reach the bottom during the morning peak, and start to swing at the middle speed back-and-forth. The traffic speeds in Group 3 have some slight resemblances to those in Group 1 during the evening peak period. However, they stabilize at the middle speed after six o'clock in the morning.

C. Input Interval Confirmation

This section investigates the effect of input interval on the prediction performance and determines the threshold p of the ACF defined in Section VI. The LSTM is employed to predict the next five-minute speed under different input intervals l over 3 random segments. From the performance listed in Table II, it can be seen that the MRE of training increases with decreasing in l. This is intuitive that small intervals can provide much historical information. However, the improvements of training MRE are insignificant when $l \leq 5$. For example, the training MRE is 3.7% and 4.4% when $l = 1$ and $l = 5$, respectively. Besides, the testing MRE at $l = 1$ is slightly larger than that at $l = 5$. The reason is that the capacity of prediction model improves with the decreasing interval, leading to overfitting. From this result, the threshold is empirically set as 0.8. For all other simulations, the input interval is set to be 5 corresponding to twenty-five minutes and thus the length of inputs is $\lceil 288/5 \rceil = 58$.

D. STTP in Network

In this section, the STTP performances of the proposed GBMs in the large-scale road network is analyzed. For comparison, one WBM and 27 IBMs are built for 27 segments, respectively. All prediction models are trained with the same configurations of LSTM. The performances of WBM, IBM and GBM are listed in Table III, where different prediction horizons N_{output} are considered.

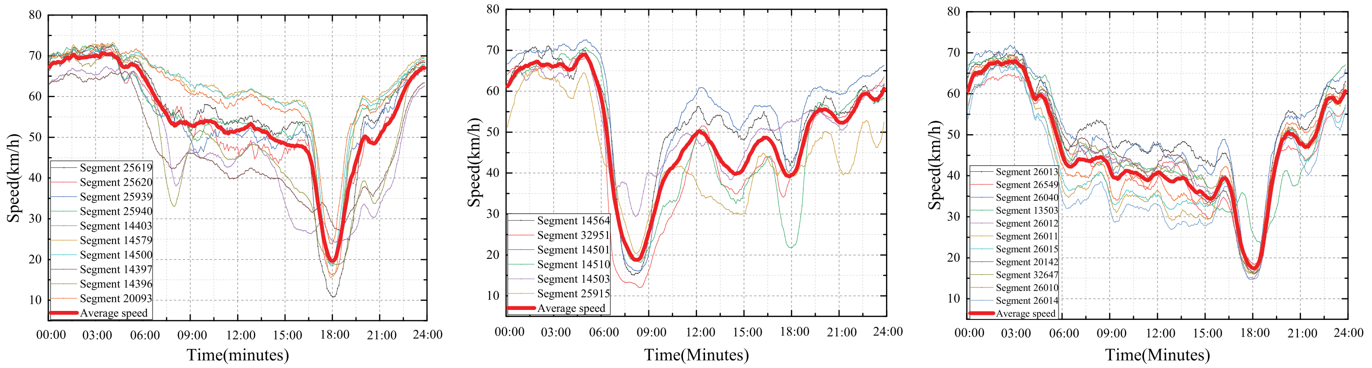
From Table III, it is intuitively obvious that the MRE increases with increasing prediction horizon. Among three prediction models, the IBMs obtain the lowest training MRE, since larger datasets are harder to fit [56]. However, GBMs can obtain lower gaps between training MREs and testing MREs than those of IBMs in all tests, since increasing the number and diversity of the training samples can improve generalization capability of the prediction model [49]. On the contrary, IBMs are constrained by the problem of overfitting resulted from modeling the noise. Fig. 15 evaluates the average prediction performances of the network. The gaps between training MREs and testing MREs of GBMs are close to be 0, while the ones of IBMs are around 2%, which further indicates that GBMs have better generalization capacity than IBMs.

As shown in Table III and Fig. 15, WBMs perform worst among three prediction models in terms of MRE in all cases. For the 5, 10 and 15-minute prediction results, the network training MRE of WBMs declines by 2%, 3% and 5%, respectively, compared to the network training MRE of IBMs. Moreover, the testing MARE of WBMs reaches up to 15% at $N_{\text{output}} = 3$. Both network and group performances indicate that the WBMs fail to model the diverse traffic patterns in large-scale networks.

From Table III, it can be observed that GBMs perform better than IBMs in terms of testing MRE in a relatively simple task of five-minute forecasting. The testing MREs of the GBMs and IBMs are 4.12% and 5.05% for Group 1, 4.07% and 4.94% for Group 2, 5.00% and 5.37% for Group 3, respectively. Moreover, GBMs achieve the lowest MIRE of 2.7% at $N_{\text{output}} = 1$. However, as the task becomes complex, the capacity of GBMs becomes insufficient. For example, the testing MREs of GBMs are around 1% more than those of IBMs when $N_{\text{output}} = 2$, while the testing MREs of GBMs are around 2% more than those of IBMs when $N_{\text{output}} = 3$.

As shown in Fig. 16, the GBMs can predict the trends of traffic speeds well, but the prediction performance gets worse with increasing prediction horizon. It can be seen that GBM of 5-minute forecasting can efficiently predict the sudden speed changes, while GBM of 10 or 15 minutes forecasting has a

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.



(a) Group 1 (b) Group 2 (c) Group 3

Fig. 14. Average traffic speeds of the road segments on weekdays with five-minute time interval from September, 2017 to November, 2017 in different groups. The thicker red lines represent the centers of the corresponding clusters.

TABLE III
THE GROUP PERFORMANCE OF THE PROPOSED FRAMEWORK

Prediction Horizon	Group	Algorithm	MRE of Training(%)	MRE of Testing(%)	Gap (%)	MARE of Testing(%)	MIRE of Testing(%)
1 (five-minute)	**1**	WBM	5.55	5.57	0	9.69	3.81
		GBM	3.97	**4.12**	0.1	6.87	**2.65**
		IBM	3.20	5.05	1.9	9.80	3.03
	2	WBM	4.81	4.86	0.1	5.58	4.20
		GBM	4.08	**4.07**	0	5.76	3.39
		IBM	3.59	4.94	1.3	6.04	3.67
	3	WBM	5.98	5.36	0.6	10.64	3.73
		GBM	4.96	**5.00**	0	6.54	4.39
		IBM	3.72	5.37	1.6	6.86	4.40
2 (Ten-minute)	**1**	WBM	7.35	7.70	0.4	10.42	4.05
		GBM	5.92	6.04	0.1	10.07	3.70
		IBM	3.80	**5.77**	2.0	9.94	3.57
	2	WBM	6.97	7.05	0.1	8.30	5.85
		GBM	6.22	6.22	0	9.86	5.29
		IBM	3.90	**5.67**	1.8	6.70	4.84
	3	WBM	7.46	7.13	0.3	11.89	5.55
		GBM	7.16	7.24	0	9.56	6.27
		IBM	4.20	**6.17**	2.0	7.73	4.91
3 (Fifteen-minute)	**1**	WBM	9.35	9.46	0.1	13.50	6.33
		GBM	7.08	7.35	0.3	11.82	4.61
		IBM	4.01	**5.82**	1.8	9.21	3.97
	2	WBM	9.62	9.68	0.1	12.02	8.32
		GBM	7.71	7.93	0.2	11.54	6.68
		IBM	4.12	**5.66**	1.5	7.00	4.69
	3	GBM	9.20	9.87	0.7	**14.94**	7.71
		GBM	8.36	8.43	0.1	12.00	7.07
		IBM	4.71	**6.38**	1.7	8.49	4.87

GBM: Group-based Model. IBM: Individual-based Model. WBM: Whole-based Model.

delayed reaction in rush hours (the dash area in Fig. 16), in which the traffic speed switches sharply. During the evening rush hour from around 4:30PM to 7:30PM, GBM predicts the next 5-minute speed with a small error rate at the beginning and the ending of congestion, while 10 or 15-minute predictions delay (with respect to the true speed). This is because that the observations made in the immediate past are usually a good indication of the short-term future. The closer the inputs to the predicted point are, the more information about the predicted point the model has. Therefore, GBMs under all prediction horizons can keep up with the trends of traffic speed well, when the speed steadily changes. However, when the speed changes

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

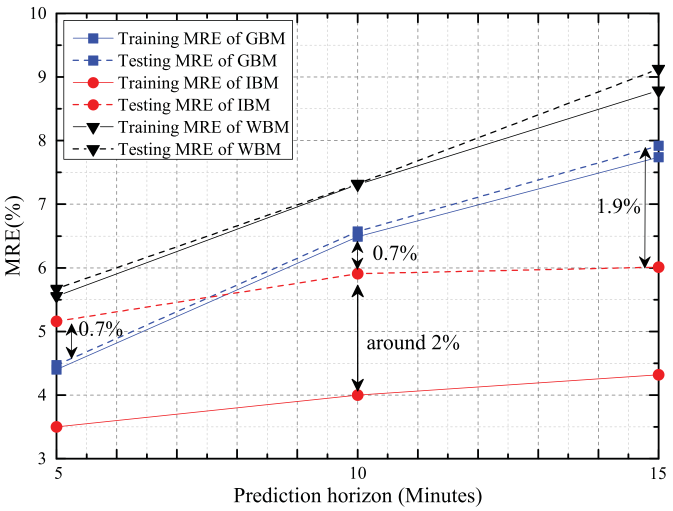


Fig. 15. Network MREs of GBM, WBM and IBM.

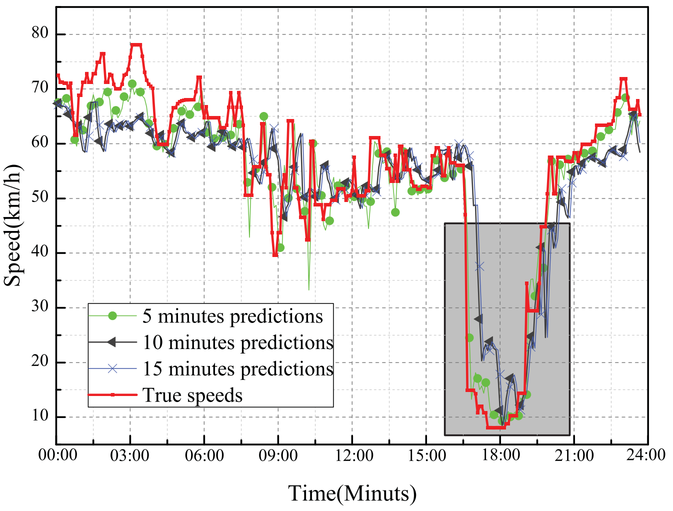


Fig. 16. The estimated speeds with different prediction horizons by GBMs versus true speeds of a random road segment in Group 1 on November 31, 2017.

sharply and suddenly, the inputs for predicting the next 10 or 15 minutes speed have less indications of such changes than those for predicting the next 5 minutes speed. In such case, the prediction model cannot capture the speed changes at the very beginning, but can adjust itself shortly after it takes the changed speed into account.

The proposed framework is scalable that can be easily applied for large-scale networks by significantly reducing the number of prediction models. It can reach the compromise between the number of prediction models and prediction performance. Compared to the traditional 27 IBMs, the number of GBMs is reduced by $\frac{(27-3)}{27} \approx 88\%$ with about $0.7\% - 1.9\%$ performance degradation, in terms of network MRE, as shown in Fig. 15. In conclusion, the prediction accuracy of the proposed framework is comparable to that of customized IBMs, which validates its ability for STTP in large-scale networks.

The prediction performance is validated over 27 road segments and may vary from the size of dataset. Besides, the reduction in the amount of the prediction model depends much on the similarity between road segments. If all road segments follow similar patterns, the GBMs degenerate into the WBMs. If each road segment belongs to a category of its own, the GBMs become the IBMs.

VIII. Conclusion

The characteristics of the multiplicity and heterogeneity make STTP in large-scale networks a challenging and important problem. By exploiting the periodicity of traffic patterns, a DL framework for STTP in large-scale networks is proposed in this paper. The key point of the framework is the combination of the DeepCluster and the DeepPrediction, as well as the model sharing strategy. The proposed framework is evaluated over a real large-scale network of Liuli Bridge in Beijing and some insights into generic DL models are obtained. Despite the prediction performances of the GBMs are slightly worse than those of IBMs in most tests, the GBMs have a better generalization ability. For five-minute prediction, the GBM obtains 0.7% error lower than IBM. The effect of input intervals on the prediction performance is also discussed, which guides the framework to select the effective input interval. Furthermore, only 3 prediction models are used to achieve STTP in a network, while the traditional way needs 27 prediction models.

Acknowledgment

The authors would like to thank Huimin Wen, Kai Xian and Zhongyuan Zhu, from Beijing Traffic Development Research Institute for sharing their traffic datasets with us during the course of this research.

References

[1] M. Wang, H. Shan, R. Lu, R. Zhang, X. Shen, and F. Bai, “Real-time path planning based on hybrid-VANET-enhanced transportation system,” *IEEE Trans. Veh. Technol.*, vol. 64, no. 5, pp. 1664–1678, May 2015.

[2] K. Zheng, L. Hou, H. Meng, Q. Zheng, N. Lu, and L. Lei, “Soft-defined heterogeneous vehicular network: Architecture and challenges,” *IEEE Netw.*, vol. 30, no. 4, pp. 72–80, Aug. 2016.

[3] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato, “An intelligent traffic load prediction-based adaptive channel assignment algorithm in SDN-IoT: A deep learning approach,” *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5141–5154, Dec. 2018.

[4] E. I. Vlahogianni, J. C. Golias, and M. G. Karlaftis, “Short-term traffic forecasting: Overview of objectives and methods,” *Transport Rev.*, vol. 24, no. 5, pp. 533–557, Sep. 2004.

[5] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, “Short-term traffic forecasting: Where we are and where we’re going,” *Transp. Res. Part C: Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.

[6] A. Ermagun and D. Levinson, “Spatiotemporal traffic forecasting: Review and proposed directions,” *Transport Rev.*, vol. 38, no. 6, pp. 786–814, Feb. 2018.

[7] K. Zheng, H. Meng, P. Chatzimisios, L. Lei, and X. Shen, “An SMDP-based resource allocation in vehicular cloud computing systems,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7920–7928, Sep. 2015.

[8] K. Zheng, F. Liu, L. Lei, C. Lin, and Y. Jiang, “Stochastic performance analysis of a wireless finite-state Markov channel,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 2, pp. 782–793, Jan. 2013.

[9] I. Lana, J. D. Ser, M. Velez, and E. Vlahogianni, “Road traffic forecasting: Recent advances and new challenges,” *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 93–109, Apr. 2018.

[10] W. Xu *et al.*, “Internet of vehicles in big data era,” *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

[11] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions," *IEEE Commun. Surv. Tut.*, vol. 17, no. 4, pp. 2377–2396, Jun. 2015.

[12] N. Kato *et al.*, "The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 146–153, Jun. 2017.

[13] Z. M. Fadlullah *et al.*, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Commun. Surv. Tut.*, vol. 19, no. 4, pp. 2432–2455, May 2017.

[14] F. Liu, K. Zheng, W. Xiang, and H. Zhao, "Design and performance analysis of an energy-efficient uplink carrier aggregation scheme," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 2, pp. 197–207, Feb. 2014.

[15] B. Mao *et al.*, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[16] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Netw.*, vol. 4, no. 2, pp. 251–257, Mar. 1991.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[18] A. Bagnall, E. Keogh, S. Lonardi S, and G. Janacek, "A bit level representation for time series data mining with shape based similarity," *Data Mining Knowl. Discovery*, vol. 13, no. 1, pp. 11–40, May 2006.

[19] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining Knowl. Discovery*, vol. 26, no. 2, pp. 275–309, Mar. 2013.

[20] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering—A decade review," *Inf. Syst.*, vol. 53, pp. 16–38, Oct. 2015.

[21] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient similarity search in sequence databases," in *Proc. 4th Int. Conf. Found. Data Org. Algorithms*, Heidelberg, Germany, Jun. 1993, pp. 69–84.

[22] F. Korn, H. V. Jagadish, and C. Faloutsos, "Efficiently supporting ad hoc queries in large datasets of time sequences," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Tucson, AZ, USA, May 1997, pp. 289–300.

[23] E. J. Keogh and M. J. Pazzani, "An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback," in *Proc. 4th Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, Aug. 1998, pp. 239–247.

[24] E. Keogh, K. Chakrabarti, M. Sharad, and M. Pazzani, "Locally adaptive dimensionality reduction for indexing large time series databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Santa Barbara, CA, USA, May 2001, pp. 151–162.

[25] E. Keogh, S. Lonardi, and C. A. Ratanamahatana, "Towards parameter-free data mining," in *Proc. 10th ACM SIGMOD Int. Conf. Knowl. Discovery Data Mining*, Seattle, WA, USA, Aug. 2004, pp. 206–215.

[26] B. K. Yi, and C. Faloutsos, "Fast time sequence indexing for arbitrary Lp norms," in *Proc. 26th Int. Conf. Very Large Data Bases (VLDB)*, Cairo, Egypt, Sep. 2000, pp. 385–394.

[27] F. L. Chung, T. C. Fu, R. W. P. Luk, and V. T. Y. Ng, "Flexible time series pattern matching based on perceptually important points," in *Proc. Int. Joint Conf. Artif. Intell. Workshop (Learn. Temporal Spatial Data)*, Seattle, WA, USA, Aug. 2001, pp. 1–7.

[28] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan, "Auto-encoder based data clustering," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, vol. 8258, J. Ruiz-Shulcloper and G. Sanniti di Baja, Eds., Berlin, Germany: Springer, 2013, pp. 117–124.

[29] F. Tian, B. Gao, Q. Cui, E. Chen, and T. Liu, "Learning deep representations for graph clustering," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1293–1299.

[30] J. Xu, P. Wang, G. Tian, B. Xu, J. Zhao, and F. Wang, "Short text clustering via convolutional neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.*, Denver, CO, USA, May 2015, pp. 62–69.

[31] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. 33 rd Int. Conf. Mach. Learn.*, San Juan, PR, USA, May 2016, pp. 478–487.

[32] K. Tian, S. Zhou, and J. Guan, "DeepCluster: A general clustering framework based on deep learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Cham, Switzerland: Springer, Dec. 2017, pp. 809–825.

[33] S. Sun, R. Huang, and Y. Gao, "Network-scale traffic modeling and forecasting with graphical lasso and neural networks," *J. Transp. Eng.*, vol. 138, no. 11, pp. 1358–1367, Nov. 2012.

[34] J. Y. Yang, L. D. Chou, C. F. Tung, S. M. Huang, and T. W. Wang, "Average-speed forecast and adjustment via VANETs," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, pp. 4318–4327, Nov. 2013.

[35] B. Yu, X. L. Song, F. Guan, Z. M. Yang, and B. Z. Yao, "k-Nearest neighbor model for multiple-time-step prediction of short-term traffic condition," *J. Transp. Eng.*, vol. 142, no. 6, pp. 1–10, 2016.

[36] G. N. Polson and O. V. Sokolov, "Deep learning for short-term traffic flow prediction," *Transp. Res. Part C: Emerg. Technol.*, vol. 79, pp. 1–17, Jun. 2017.

[37] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Representation*, Vancouver, Canada, Apr. 2018.

[38] M. Ben-Akiva, M. Bierlaire, H. Koutsopoulos, and R. Mishalani, "DynaMIT: A simulation-based system for traffic prediction," in *Proc. DACCORD Short Term Forecasting Workshop*, Delft, Netherlands, Feb. 1998, pp. 1–12.

[39] T. Djukic, J. W. C. Van Lint, and S. P. Hoogendoorn, "Application of principal component analysis to predict dynamic origin-destination matrices," *Transp. Res. Rec.: J. Transp. Res. Board*, vol. 2283, no. 1, pp. 81–89, Jan. 2012.

[40] N. Mitrovic, M. T. Asif, U. Rasheed, J. Dauwels, and P. Jaillet, "CUR decomposition for compression and compressed sensing of large-scale traffic data," in *Proc. 16th Int. Conf. Intell. Transp. Syst. (ITSC)*, Hague, Netherlands, Oct. 2013, pp. 1475–1480.

[41] M. T. Asif, S. Kannan, J. Dauwels, and P. Jaillet, "Data compression techniques for urban traffic data," in *Proc. IEEE Symp. Comput. Intell. Vehicles Transp. Syst.*, Singapore, Apr. 2013, pp. 44–49.

[42] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. Part C: Emerg. Technol.*, vol. 19, no. 4, pp. 606–616, Aug. 2011.

[43] Z. Zhao, W. Chen, X. Wu, P. Chen, and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, Jan. 2017.

[44] A. Koesdwiady, R. Soua, and F. Karray, "Improving traffic flow prediction with weather information in connected cars: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9508–9517, Dec. 2016.

[45] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. P. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4. pp. 1–16, Apr. 2017.

[46] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 815–823.

[47] K. Xie *et al.*, "Accurate recovery of internet traffic data: A sequential tensor completion approach," *IEEE/ACM Trans. Netw.*, vol. 26, no. 2, pp. 793–806, Apr. 2018.

[48] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998

[49] V. N. Vapnik and A. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," *Theory Probability Appl.*, vol. 16, no. 2, pp. 264–280, 1971.

[50] C. Chen, Y. Wang, L. Li, J. Hu, and Z. Zhang, "The retrieval of intra-day trend and its influence on traffic prediction," *Transp. Res. Part C*, vol. 22, pp. 103–118, Jun. 2012.

[51] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[52] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybern.*, vol. 3, no. 3, pp. 32–57, Sep. 1973.

[53] R. B. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Statist.-Theory Methods*, vol. 3, no. 1, pp. 1–27, Sep. 1974.

[54] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 1, pp. 224–227, Apr. 1979.

[55] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.

[56] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.

Lingyi Han received the B.S. degree from the Beijing University of Posts and Telecommunications, China, in 2016. She is currently pursuing the Ph.D. degree with the Intelligent Computing and Communication Laboratory and Key Laboratory of Universal Wireless Communications, Beijing University of Posts and Telecommunications, Ministry of Education. Her research interests include datamining and artificial intelligence in the Internet of Things.

Kan Zheng (SM'09) received the B.S., M.S., and Ph.D. degrees from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1996, 2000, and 2005, respectively. He is currently a Professor with BUPT. He is the Author of more than 200 journal articles and conference papers in the field of resource optimization in wireless networks, M2M networks, VANET, and so on. He has rich industry experiences on the standardization of the new emerging technologies. Dr. Zheng holds editorial board positions for several journals. He has organized several special issues in famous journals, including the IEEE COMMUNICATIONS ON SURVEYS AND TUTORIALS and *Transactions on Emerging Telecommunications Technologies*.

Long Zhao (M'17) received the Ph.D. degree from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015, where he is currently an Associate Professor. From April 2014 to March 2015, he was a Visiting Scholar with the Department of Electrical Engineering, Columbia University. His research interests include intelligent wireless communications and massive signal processing.

Xianbin Wang (S'98–M'99–SM'06–F'17) received the Ph.D. degree in electrical and computer engineering from the National University of Singapore in 2001. He is a Professor and Tier 1 Canada Research Chair of Western University, Canada. Prior to joining Western, he was with the Communications Research Centre Canada (CRC) as a Research Scientist/Senior Research Scientist between July 2002 and December 2007. From January 2001 to July 2002, he was a System Designer with STMicroelectronics. He has more than 400 peer-reviewed journal and conference papers, in addition to 30 granted and pending patents and several standard contributions. His current research interests include 5G and beyond, Internet of Things, communication security, machine learning, and intelligent communications. Dr. Wang is a Fellow of Canadian Academy of Engineering and an IEEE Distinguished Lecturer. He has received many awards and recognitions, including the Canada Research Chair, CRC Presidents Excellence Award, Canadian Federal Government Public Service Award, Ontario Early Researcher Award, and six IEEE best paper awards. He is currently serving as an Editor/Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON BROADCASTING, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He was also an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS between 2007 and 2011 and for the IEEE WIRELESS COMMUNICATIONS LETTERS between 2011 and 2016. He was involved in many IEEE conferences, including GLOBECOM, ICC, VTC, PIMRC, WCNC, and CWIT, in different roles such as Symposium Chair, Tutorial Instructor, Track Chair, Session Chair, and TPC Co-Chair. He is currently serving as the Chair of the ComSoc SPCE Technical Committee and a member of the IEEE Fellow Committee.

Xuemin (Sherman) Shen (M'97–SM'02–F'09) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on resource management, wireless network security, social networks, 5G and beyond, and vehicular ad hoc and sensor networks. He is a Registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. Dr. Shen received the R.A. Fessenden Award in 2019 from IEEE, Canada, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015, and the Education Award in 2017 from the IEEE Communications Society. He has also received the Excellent Graduate Supervision Award in 2006 and the Outstanding Performance Award five times from the University of Waterloo and the Premiers Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for the IEEE Globecom16, IEEE Infocom14, IEEE VTC10 Fall, and IEEE Globecom07, the Symposia Chair for the IEEE ICC10, the Tutorial Chair for the IEEE VTC11 Spring, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the Editor-in-Chief for the IEEE INTERNET OF THINGS JOURNAL and the Vice-President on Publications of the IEEE Communications Society.

Authorized licensed use limited to: University of Roehampton. Downloaded on June 25,2024 at 15:32:52 UTC from IEEE Xplore. Restrictions apply.