

# Spatiotemporal Patterns in Large-Scale Traffic Speed Prediction

Muhammad Tayyab Asif, *Student Member, IEEE*, Justin Dauwels, *Senior Member, IEEE*, Chong Yang Goh, Ali Oran, Esmail Fathi, Muye Xu, Menoth Mohan Dhanya, Nikola Mitrovic, and Patrick Jaillet

**Abstract**—The ability to accurately predict traffic speed in a large and heterogeneous road network has many useful applications, such as route guidance and congestion avoidance. In principle, data-driven methods, such as support vector regression (SVR), can predict traffic with high accuracy because traffic tends to exhibit regular patterns over time. However, in practice, the prediction performance can significantly vary across the network and during different time periods. Insight into those spatiotemporal trends can improve the performance of intelligent transportation systems. Traditional prediction error measures, such as the mean absolute percentage error, provide information about the individual links in the network but do not capture global trends. We propose unsupervised learning methods, such as  $k$ -means clustering, principal component analysis, and self-organizing maps, to mine spatiotemporal performance trends at the network level and for individual links. We perform prediction for a large interconnected road network and for multiple prediction horizons with an SVR-based algorithm. We show the effectiveness of the proposed performance analysis methods by applying them to the prediction data of the SVR.

**Index Terms**—Large-scale network prediction, spatiotemporal error trends.

## I. INTRODUCTION

INTELLIGENT transport systems (ITSs) can provide enhanced performance by incorporating data that are related to the future state of road networks [1]. Traffic prediction is useful for many applications such as route guidance and congestion avoidance [1]–[3]. Traffic prediction requires learning

the nonlinear relationships between past and future traffic states [2], [3]. Data-driven methods, such as support vector regression (SVR), tend to provide better prediction results than competing methods [2]–[9]. However, these studies usually consider scenarios such as expressways or a few intersections. Studies such as the works in [2], [3], and [10]–[12] only consider highways or motorways. Some other studies, such as the works in [6] and [8], do consider more general scenarios, albeit for some small regions. Consequently, the performance patterns in large heterogeneous networks have not been investigated. Min and Wynter considered a moderate-sized road network that consists of about 500 road segments [13]. However, they developed a custom model for the test area, which is not available. This limits any meaningful comparison with their proposed method. They analyzed the performance by taking into account different road categories. We will also consider road-category-wise comparison as one of the indexes for performance evaluation. Traffic prediction for large networks requires modular and easily scalable algorithms. The methods should also provide accurate prediction for multiple prediction horizons. In this paper, we analyze the performance of data-driven methods, such as the SVR, for large-scale prediction. The network comprises of roads with different speed limits and capacities, and it covers different areas (urban, rural, and downtown).

Traffic prediction studies usually consider point estimation methods that are similar to the mean absolute percentage error (MAPE) to analyze prediction performance [2]–[11], [13], [14]. These measures work well for overall performance comparison. However, they fail to provide any insight into underlying spatiotemporal prediction patterns. Forecasting methods may not provide uniform prediction performance across the road network. Moreover, prediction accuracy may also depend upon the time and the day. These spatiotemporal performance trends contain useful information about the predictability of the network. ITS applications can provide more robust and accurate solutions by utilizing such trends.

We apply a temporal-window-based SVR method to perform a large-scale traffic prediction. For analysis, we consider a large road network from Outram Park to Changi in Singapore. The road network consists of more than 5000 road segments. We compare the performance of the SVR with other commonly used time-series prediction algorithms, such as the artificial neural networks (ANNs) and exponential smoothing. We also provide performance comparison for different road categories in this paper. To extract spatiotemporal prediction patterns for large networks, we propose unsupervised learning methods such as the  $k$ -means clustering and the principal component

Manuscript received May 8, 2013; revised September 12, 2013; accepted October 22, 2013. Date of publication December 3, 2013; date of current version March 28, 2014. This work was supported in part by the Singapore National Research Foundation through the Singapore–Massachusetts Institute of Technology Alliance for Research and Technology (SMART) Center for Future Mobility (FM). This paper was presented in part at the IEEE Intelligent Transportation Systems Conference, Anchorage, AK, USA, 2012. The Associate Editor for this paper was S. C. Wong.

M. T. Asif, J. Dauwels, M. M. Dhanya, and N. Mitrovic are with the School of Electrical and Electronic Engineering, College of Engineering, Nanyang Technological University, Singapore (e-mail: muhammad89@e.ntu.edu.sg; jdauwels@ntu.edu.sg).

C. Y. Goh is with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA.

A. Oran is with Center for Future Urban Mobility, Singapore-MIT Alliance for Research and Technology, Singapore (e-mail: aoran@smart.mit.edu).

E. Fathi is with Siemens Private Ltd., Singapore.

M. Xu is with the Trading Technology Group, Deutsche Bank AG, Singapore

P. Jaillet is with the Department of Electrical Engineering and Computer Science, School of Engineering, and also with the Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jaillet@mit.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TITS.2013.2290285

analysis (PCA). For link-level temporal prediction patterns, we use self-organizing maps (SOMs). We apply these data-mining algorithms to extract performance trends in the SVR prediction data.

This paper is structured as follows. In Section II, we briefly discuss the problem of large-scale prediction. In Section III, we propose different data-driven algorithms for large-scale prediction. In Section IV, we explain the data set and performance measures for comparison. In Section V, we compare the prediction performance of the SVR with that of other methods. In Section VI, we develop unsupervised learning techniques for extracting spatiotemporal performance trends in large-scale prediction. In Section VII, we evaluate the efficiency of the proposed data-mining methods with the prediction data of the SVR. Finally, in Section VIII, we summarize our contributions, and we suggest topics for future work.

## II. LARGE-SCALE PREDICTION

In this section, we briefly discuss the problem of large-scale prediction. We also discuss the selection of training and test data for the supervised learning algorithms.

### A. Traffic Prediction

We represent the road network by a directed graph  $G = (N, E)$ . Set  $E$  contains  $p$  road segments (links)  $\{s_i\}_{i=1}^p$ . Space averaged speed value  $z(s_i, t_j)$  represents the weight of link  $s_i$  during interval  $(t_j - \delta_t, t_j)$ . Sampling interval  $\delta_t$  is 5 min. Future traffic trends strongly depend on the current and past behavior of that road and its neighbors [2], [3], [13]. Suppose  $\{\theta_u \in \Theta_{s_i}\}_{u=1}^l$  is the set of road segments containing  $s_i$  and its neighbors, such that  $\Theta_{s_i} \subseteq E$ . Our aim will be to find the relationship function  $f$  between the current/past traffic data  $\{z(\theta_u, t_j - q\delta_t) | u = 1, \dots, l, q = 0, \dots, m_{\theta_u}\}$  and the future traffic variations  $\hat{z}(s_i, t_j + k\delta_t)$  such that

$$\hat{z}(s_i, t_j + k\delta_t) = f(z(\theta_1, t_j), \dots, z(\theta_l, t_j - m_{\theta_l}\delta_t)). \quad (1)$$

Feature set  $\{m_{\theta_u}\}_{u=1}^l$  determines the horizon of the past speed values of link  $\theta_u$ , which are used to predict the  $k$ -step ahead speed values of  $s_i$ . We will refer to the  $k$ -step ahead prediction as the  $k^{\text{th}}$  prediction horizon.

We need to determine relevant links  $\Theta_{s_i}$  (the spatial features) and time lags  $m_{\theta_u}$  (the temporal features) to predict  $\hat{z}(s_i, t_j + k\delta_t)$ . Extracting the spatial features is a computationally expensive task [15]. This additional computational cost severely limits the scalability of the prediction algorithm for large and generic road networks. Therefore, we will not consider spatial features in this paper. For large-scale prediction, we consider the following variant of (1), which is termed as the temporal window method [6]:

$$\hat{z}(s_i, t_j + k\delta_t) = f(z(s_i, t_j), \dots, z(s_i, t_j - m_{s_i}\delta_t)). \quad (2)$$

In (2), we only consider the past historical trends of  $s_i$  to predict  $\hat{z}(s_i, t_j + k\delta_t)$ . The temporal window method for feature selection has been demonstrated to effectively work for data-driven

traffic prediction algorithms [3], [4], [6], [7], [9], [16]. Different methods have been proposed to take further advantage of inherent temporal traffic patterns for enhanced prediction accuracy [8], [10], [12], [17]–[21]. These methods employ different feature selection techniques to prepartition the data according to temporal patterns (time of the day, weekdays/weekends, etc.). The feature selection algorithms include SOMs [12], [18], [21], genetic algorithms [8], [19], wavelets [20], and committees of neural networks [10]. Another proposed method combines the Kalman filter with the seasonal autoregressive integrated moving average (SARIMA) [12], [17]. These techniques, however, are computationally expensive, which limits their scalability for large networks. Furthermore, Lippi *et al.* showed that the traditional SVR can provide similar performance to these ensemble methods without suffering from extra computational overhead [12]. Consequently, we will consider the SVR with temporal window for large-scale prediction. We train separate predictors for each link  $s_i$  and each prediction horizon  $k$ . The temporal window method for feature selection allows predictors from different links and prediction horizons to run in parallel. Furthermore, these algorithms are independent of each other. Therefore, they can efficiently run on distributed platforms with minimum communication overhead.

### B. Training and Test Data for Supervised Learning

Supervised learning methods, such as the SVR and ANNs, assume that the labeled training data and the test data come from the same distribution [22]–[24]. Hence, it is unnecessary to retrain the algorithm every time new data become available. Traffic prediction methods also follow the same assumption [2]–[9], [13], [16], [25]. Similar to other studies, we train the algorithm with 50 days of data and perform prediction for 10 days [3], [5], [13]. It is important to point out that this assumption may not hold true in the long term. Factors such as changes in transportation infrastructure, residential location, fuel prices, and car ownership can significantly affect long-term traffic patterns [25], [26]. Supervised learning methods may not work well in such cases. Techniques that are based on transfer learning might prove useful in such scenarios [24].

## III. TRAFFIC PREDICTION ALGORITHMS

We apply the SVR to perform a large-scale prediction. We briefly explain the algorithm in this section. We compare the performance of the SVR with those of the ANN and exponential smoothing. We also briefly discuss these algorithms in this section.

### A. SVR

The SVR is a data-driven prediction algorithm. It is commonly employed for time-series prediction [27]. With the temporal feature selection, the input feature vector  $\mathbf{x}_j \in \mathbb{R}^n$  at time  $t_j$  for link  $s_i$  will be  $\mathbf{x}_j = [z(s_i, t_j), \dots, z(s_i, t_j - m_{s_i}\delta_t)]^T$ . Feature vector  $\mathbf{x}_j$  contains the current average speed of the road  $z(s_i, t_j)$  and the  $m_{s_i}$  past speed values. Let  $y_{jk} = z(s_i, t_j + k\delta_t)$  be the future speed value at time  $t_j + k\delta_t$ . We aim to find

the relationship between  $y_{jk}$  and  $\mathbf{x}_j$ . To this end, we use the historical speed data of  $s_i$  to train the SVR. The training data contains  $r$  2-tuples  $\{(\mathbf{x}_j, y_{jk})\}_{j=1}^r$ . We use the SVR to infer the nonlinear relationships between  $\mathbf{x}_j$  and  $y_{jk}$  and to find  $f_k$  in (2) for the  $k$ th prediction horizon.

We briefly explain the SVR algorithm here. More rigorous treatment of the topic can be found in [22] and [23]. Let us consider the formulation of the SVR, which is called  $\varepsilon$ -SVR, formulated as [22]

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{w} \cdot \mathbf{w} + C \sum_{j=1}^r (\xi_j + \xi_j^*) \\ & \text{subject to} && \begin{cases} y_{jk} - \mathbf{w} \cdot \mathbf{x}_j - b \leq \varepsilon + \xi_j \\ \mathbf{w} \cdot \mathbf{x}_j + b - y_{jk} \leq \varepsilon + \xi_j^* \\ \xi_j, \xi_j^* \geq 0 \end{cases} \end{aligned} \quad (3)$$

where  $\mathbf{w}$  is the required hyperplane, and  $\xi_j$  and  $\xi_j^*$  are the slack variables. It uses the so-called insensitive loss function, which imposes cost  $C$  on training points having deviations of more than  $|\varepsilon|$ . It is often hard to predefine the exact value of error bound  $\varepsilon$  [28]. This problem can be avoided by adopting a variant of the SVR, which is called the  $\nu$ -SVR [28]. Hence, we will employ the  $\nu$ -SVR to perform speed prediction. The SVR nonlinearly maps (not explicitly) the input speed data into some higher dimensional feature space  $\Phi$  [22], [28]. It then finds the optimal hyperplane in that high-dimensional feature space  $\Phi$ . The kernel trick helps the SVR to avoid this explicit mapping in  $\Phi$ . Let us choose  $\kappa$  as the desired kernel function. Then, we can replace the dot products in the feature space by the relation  $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$  [22]. Function  $f_k$  will be [22], [28]

$$f_k(\mathbf{x}) = \sum_{j=1}^r (\alpha_j - \alpha_j^*) \kappa(\mathbf{x}, \mathbf{x}_j) + b \quad (4)$$

where  $\alpha_j$  and  $\alpha_j^*$  are the Lagrange multipliers. We employ (4) to train the SVR and to perform the speed prediction. The Matlab package LIBSVM is used for the SVR implementation [29].

For this paper, we consider the radial basis function kernel. It is highly effective in mapping nonlinear relationships [30]. Consequently, it is commonly employed for performing traffic prediction [3], [4].

## B. ANNs

ANNs can perform time-series prediction with high accuracy [31]. Consequently, they have been extensively used for traffic parameter forecasting in different configurations [2], [4], [6], [8], [10], [11], [16], [32]. Multilayer feedforward neural networks are the most commonly employed configuration for traffic prediction [4], [16], [32].

We apply a feedforward neural network for the large-scale speed prediction of network  $G$  across multiple prediction horizons. We consider the temporal window for the feature selection for the ANN. We apply backpropagation to train the ANN. We train separate neural networks for different links and prediction horizons.

TABLE I  
CATEGORIES OF ROAD SEGMENTS

Category	CATA	CATB	CATC	Slip Roads	Other
No. of links	703	2818	841	592	70

## C. Exponential Smoothing

Exponential smoothing is a commonly employed method to perform time-series prediction. It is also applied for traffic parameter prediction [33]. The prediction is computed as a weighted average of past data points. Specifically, weights of past values exponentially decay with decay factor  $\chi_k$  for the  $k$ th prediction horizon.

## IV. DATA SET AND PERFORMANCE MEASURES

In this section, we explain the data set for the performance evaluation. We consider a large subnetwork in Singapore, which covers the region from Outram Park to Changi. The road network  $G$  consists of a diverse set of roads having different lane counts, speed limits, and capacities. It includes three expressways, i.e., East Coast Parkway, Pan Island Expressway, and Kallang–Paya Lebar Expressway. The network also includes areas carrying significant traffic volumes, such as Changi Airport and the central business district.

Overall, network  $G$  consists of  $p = 5024$  road segments. These road segments are grouped into different categories by the Land Transport Authority (LTA) of Singapore. Table I shows the number of road segments for each category in  $G$ . In this paper, we consider the speed data provided by the LTA. The data set has an averaging interval of 5 min. We choose the speed data from the months of March and April 2011.

We now explain the performance measures that are used to assess the prediction accuracy of the proposed algorithms. We calculate the absolute percentage error (APE) for  $s_i$  at time  $t_j$  for  $k$ th prediction horizon  $e(s_i, k, t_j)$  as follows:

$$e(s_i, k, t_j) = \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)}. \quad (5)$$

The MAPE for link  $s_i$  and  $k$ th prediction horizon  $e_s(s_i, k)$  is defined as

$$e_s(s_i, k) = \frac{1}{d} \sum_{j=1}^d \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)} \quad (6)$$

where  $d$  is the number of test samples, and  $\hat{z}_k(s_i, t_j)$  is the predicted speed value at time  $t_j$  for the  $k$ th prediction horizon. The MAPE is a commonly used metric to assess the accuracy of traffic prediction algorithms [3], [4], [7], [14]. For the whole network  $G$  containing  $p$  links, we calculate the MAPE  $e_G(k)$  for the  $k$ th prediction horizon as

$$e_G(k) = \frac{1}{p} \sum_{i=1}^p e_s(s_i, k). \quad (7)$$

We calculate the standard deviation (SD) of error  $\sigma_k$  for each prediction horizon as

$$\sigma_k = \sqrt{\frac{1}{p} \sum_{i=1}^p (e_s(s_i, k) - e_G(k))^2}. \quad (8)$$

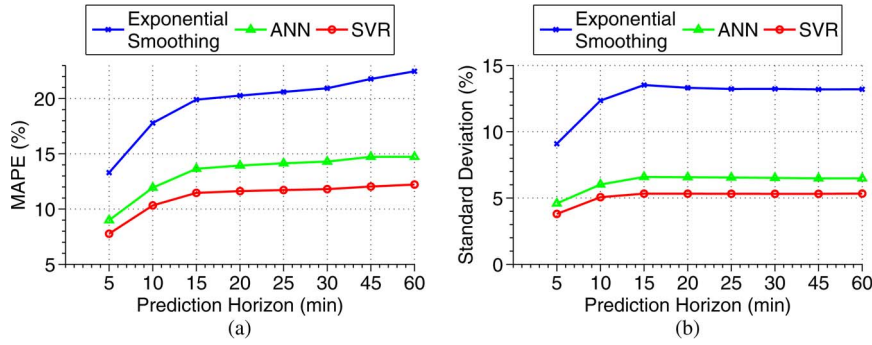


Fig. 1. Performance comparison of the prediction algorithms for different prediction horizons. (a) Mean error for the different algorithms. (b) Error SD within the network.

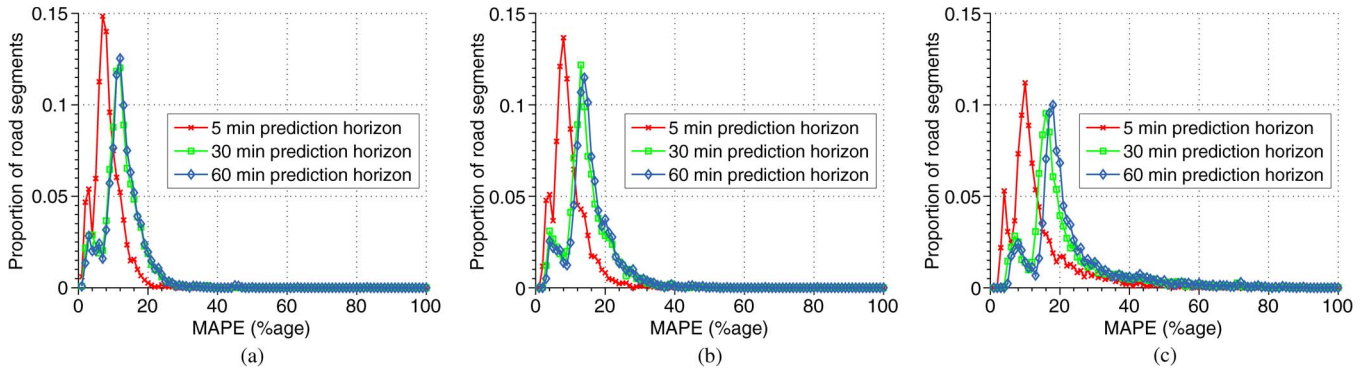


Fig. 2. Error distribution of the algorithms for different prediction horizons. (a) Error distribution of the SVR. (b) Error distribution of the ANN. (c) Error distribution of the exponential smoothing.

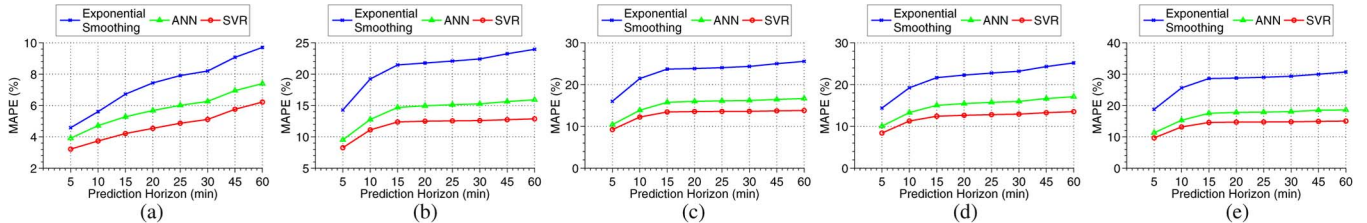


Fig. 3. Road-category-wise performance comparison. (a) CATA. (b) CATB. (c) CATC. (d) Slip roads. (e) Other categories.

We use these measures to assess the performance of the SVR, ANN, and exponential smoothing models for large-scale prediction.

### V. COMPARISON OF DIFFERENT PREDICTION ALGORITHMS

In this section, we compare the prediction performance of the SVR with that of the ANN and that of the exponential smoothing. Fig. 1 provides the network-level comparison of the performance of the algorithms. Fig. 2 shows the distribution of prediction error across different horizons. Figs. 3 and 4 show the performance of the proposed methods for different road categories.

The SVR has the smallest MAPE across different prediction horizons for the whole network  $G$  [see Fig. 1(a)]. It also has the smallest SD of error between different links [see Fig. 1(b)]. The ANN provides a slightly larger error compared with the SVR. This can be attributed to the problem of local minima that is associated with ANN training algorithms [34]. Overall,

the prediction performance for all three algorithms degrades as the prediction horizon increases. Performance degradation tends to flatten out, particularly for the data-driven methods (the SVR and the ANN) for large prediction horizons [see Figs. 1(a) and 3].

Error distribution plots (see Fig. 2) show variations in the prediction error across the road network. We observe that the distribution of prediction error (MAPE) varies from one prediction horizon to another. We also observe more than one peak for each prediction horizon. This implies that there might exist different groups of links with similar prediction performance. Let us analyze the prediction behavior of different road categories.

Figs. 3 and 4 show how the prediction error and the standard deviation vary from one road category to another. The SVR still provides the lowest MAPE (see Fig. 3) and error standard deviation [see Fig. 4, except for Category A (CATA), as shown in Fig. 4(a)]. As expected, expressways are relatively easy to predict compared with other road categories. This can be verified by comparing the performance of the predictors for

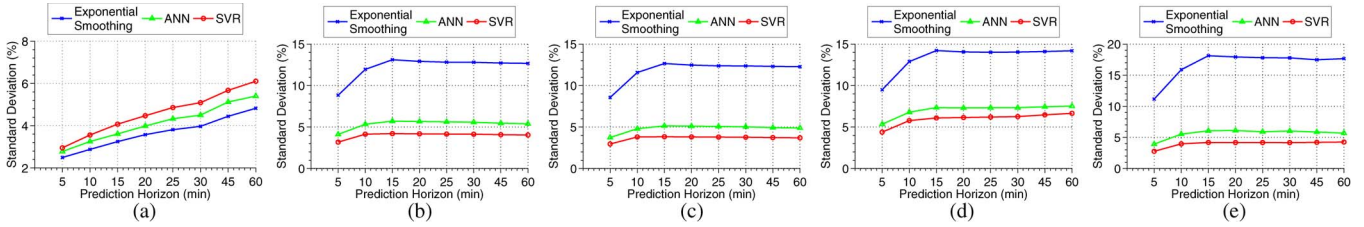


Fig. 4. Standard deviation of the error within each road category. (a) CATA. (b) CATB. (c) CATC. (d) Slip roads. (e) Other categories.

CATA roads [expressways, see Figs. 3(a) and 4(a)] with that of the other categories. However, we still observe a high SD of error within the expressways, particularly for large prediction horizons. We find similar patterns for other road categories. Overall, the SD of error within different road categories is not that different from the network-wide SD [see Figs. 1(b) and 4]. We find that the roads that belong to the same category can still have dissimilar prediction performance.

The point estimation methods work well to evaluate and compare the prediction performance of different methods. However, they provide little insight into the spatial distribution of performance patterns. For instance, we fail to identify which set of links provide worse prediction performance and vice versa. Moreover, we cannot extract the temporal performance patterns across the network and for individual roads.

In the following section, we propose unsupervised learning methods to analyze these trends.

## VI. SPATIOTEMPORAL PATTERNS IN PREDICTION PERFORMANCE

In this section, we consider the prediction data analysis as a data-mining problem. For this purpose, we propose unsupervised learning methods to find spatiotemporal performance patterns in a large-scale prediction. We analyze the efficiency of the proposed algorithms by applying them to the prediction data of the SVR.

Traffic prediction studies commonly employ measures such as the MAPE to evaluate the performance of the algorithms [2]–[11], [13], [14], [16]. These measures are inadequate to provide any information about the underlying spatiotemporal behavior of prediction algorithms. If  $e_s(s_i, k)$  represents the mean prediction error that is observed for  $s_i$ , then MAPE  $e_G(k)$  across the test network  $G$ , in a more convenient form, will be

$$e_G(k) = \frac{1}{pd} \sum_{i=1}^p \sum_{j=1}^d \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)}. \quad (9)$$

In (9), we obtain the averaged effect of errors across different links and during different time periods. Consider a large network  $G$  containing thousands of links and prediction performed for multiple prediction horizons. The prediction error of a particular link  $s_i$  at time  $t_j$  might be different from that at  $t_{j'}$ . It might vary from day to day or change during different hours. Similarly, the prediction performance between any two links  $s_i$  and  $s_j$  may also vary significantly. However, in (9) all these trends are averaged. We observed earlier that the prediction

performance may not remain uniform across large networks (see Figs. 1–4). We also observed that the point estimation methods provide little detail about these spatial variations. Moreover, these measures do not give any information about temporal performance variations.

The spatiotemporal patterns provide insight about long-term predictability and short-term predictability. Hence, they can be highly useful for ITS applications such as route guidance and traffic management.

For analysis, we consider three components of (9), which are space  $s_i$ , time  $t_j$ , and prediction horizon  $k$ . We perform cluster analysis to obtain the spatial prediction patterns. This will help find the roads with similar overall performance across different prediction horizons. To find the temporal performance patterns, we combine the PCA with the  $k$ -means clustering. The temporal patterns help us identify roads with variable and consistent prediction performance during different time periods. We also analyze daily and hourly performance patterns for individual links by applying the SOM.

### A. Analysis of Spatial Prediction Patterns

In this section, we propose the  $k$ -means clustering to find the spatial prediction patterns. This method creates different groups (clusters) of road segments. We represent these clusters by labels  $\{\omega_i\}_{i=1}^w$ . Each group (cluster) contains roads that provide similar prediction performance across different prediction horizons. To compare the links, we represent each link  $s_i$  by a vector  $\mathbf{e}_{s_i} = [e_s(s_i, 1), \dots, e_s(s_i, k)]^T$ , where  $e_s(s_i, k)$  is the MAPE for the  $k$ th prediction horizon for  $s_i$ . The distance measure  $\Delta_s(s_i, s_j)$  between links  $s_i$  and  $s_j$  is defined as

$$\Delta_s(s_i, s_j) = \sqrt{(\mathbf{e}_{s_i} - \mathbf{e}_{s_j})^T (\mathbf{e}_{s_i} - \mathbf{e}_{s_j})}. \quad (10)$$

We apply an unsupervised clustering method, as no prior knowledge about the groups of links  $\{\omega_i\}_{i=1}^w$  is available. An unsupervised learning approach creates clusters of links depending on their predictability (mean prediction error).

We use the  $k$ -means clustering to find the roads with similar performance [35]. For the  $k$ -means clustering, we need to specify the number of the clusters  $w$  beforehand. However, this information is not available for network  $G$ . We require the clusters to be composed of roads with similar performance. Moreover, the performance of the different clusters should be different from one another. This way, we expect to have a cluster of roads with high prediction accuracy and vice versa. This problem is usually referred to as the cluster validation [36], [37].

We consider commonly applied cluster validation techniques such as the Silhouette index [38], the Hartigan index [39], and the homogeneity and separation index [36], [37] in this paper.

The Silhouette index  $\Psi_{\text{sil}}(s_i)$  for link  $s_i$  belonging to cluster  $\omega_j$  is defined as

$$\Psi_{\text{sil}}(s_i) = \frac{\beta_2(s_i, \omega'_j) - \beta_1(s_i)}{\max(\beta_1(s_i), \beta_2(s_i, \omega'_j))} \quad (11)$$

where  $\beta_1(s_i)$  is the mean distance [in the sense of (10)] of  $s_i$  with the other links in cluster  $\omega_j$ . In (11),  $\beta_2(s_i, \omega'_j)$  is the mean distance of  $s_i$  with the links in the nearest cluster  $\omega'_j$ . We chose the clustering structure with the highest mean value  $\zeta_{\text{sil}}(w)$  such that

$$\zeta_{\text{sil}}(w) = \frac{1}{p} \sum_{s_i \in G} \Psi_{\text{sil}}(s_i). \quad (12)$$

The Hartigan index  $\zeta_{\text{har}}(w)$  for data size  $N$  is calculated as [40]

$$\zeta_{\text{har}}(w) = (N - w - 1) \frac{\Omega(w) - \Omega(w + 1)}{\Omega(w + 1)}. \quad (13)$$

It considers the change in mean intracluster dispersion  $\Omega(w)$  due to the change in the number of the clusters  $w$  [39], [40]. Consider a clustering structure with  $w$  clusters and  $\{g_i\}_{i=1}^w$  links in each cluster. The intracluster dispersion for the structure will be

$$\Omega(w) = \sum_{j=1}^w \sum_{i=1}^{g_j} \Delta_s(s_i, c_j)^2 \quad (14)$$

where  $\{c_i\}_{i=1}^w$  contains cluster centroids.

We use these indexes to select the optimal number of clusters. To this end, we require that the indexes agree upon on a certain  $w^*$ . We will treat the corresponding clustering structure  $\{\omega_i\}_{i=1}^{w^*}$  as the best model for network  $G$ .

### B. Analysis of Temporal Prediction Patterns

In this section, we propose methods to infer the variations in prediction performance during different time intervals.

The prediction error for a certain set of links  $\tau_i$  may not significantly change during different times of the day and across different days. For the other group  $\tau_j$ , the prediction performance might significantly vary from one period to another. We refer to these as the consistent and inconsistent clusters, respectively. We combine the PCA and the  $k$ -means clustering to identify the consistent and inconsistent clusters.

We also analyze the temporal performance trends for individual roads. For a given link  $s_i$ , some days (hours) will have similar performance patterns and vice versa. We apply the SOM to extract these trends.

We now explain our proposed methods to infer the network-level and link-level temporal prediction patterns.

1) *Network-Level Temporal Prediction Patterns*: We consider the variations in the prediction error of the links during

different days and hours to group them together. We use the PCA to deduce these daily and hourly performance patterns.

We define the daily  $\{\mathbf{d}_j(s_i) \in \mathbb{R}^{n_d}\}_{j=1}^{m_d}$  and hourly  $\{\mathbf{h}_l(s_i) \in \mathbb{R}^{n_h}\}_{l=1}^{m_h}$  performance patterns as follows. Vector  $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$  comprises the APE for all the time periods and prediction horizons for that day  $\{j\}_{j=1}^{m_d}$  for link  $s_i$ . Vector  $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$  contains the APE across all the days and prediction horizons for link  $s_i$  during hour  $\{l\}_{l=1}^{m_h}$ .

Daily variation matrix  $\mathbf{D}(s_i) = [\mathbf{d}_1(s_i), \dots, \mathbf{d}_{m_d}(s_i)]$  and hourly variation matrix  $\mathbf{H}(s_i) = [\mathbf{h}_1(s_i), \dots, \mathbf{h}_{m_h}(s_i)]$  contain all such patterns for link  $s_i$ . To quantify the performance variations within different days  $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$  and hours  $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$ , we construct the corresponding covariance matrices  $\Sigma_d(s_i)$  and  $\Sigma_h(s_i)$ , respectively. By centralizing  $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$  and  $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$  about their means, we obtain  $\mathbf{D}'(s_i)$  and  $\mathbf{H}'(s_i)$ , respectively. Covariance matrices  $\Sigma_d(s_i)$  and  $\Sigma_h(s_i)$  are calculated as follows:

$$\Sigma_d(s_i) = \frac{1}{n_d} \mathbf{D}'(s_i)^T \mathbf{D}'(s_i) \quad (15)$$

$$\Sigma_h(s_i) = \frac{1}{n_h} \mathbf{H}'(s_i)^T \mathbf{H}'(s_i). \quad (16)$$

The eigenvalue decomposition of the covariance matrices will yield

$$\Sigma_d(s_i) = \mathbf{U}_d(s_i) \Lambda_d(s_i) \mathbf{U}_d(s_i)^T \quad (17)$$

$$\Sigma_h(s_i) = \mathbf{U}_h(s_i) \Lambda_h(s_i) \mathbf{U}_h(s_i)^T \quad (18)$$

where matrices  $\{\mathbf{U}_j(s_i) = [\varphi_{j1}(s_i), \dots, \varphi_{jm_j}(s_i)]\}_{j \in \{d, h\}}$  and  $\{\Lambda_j(s_i)\}_{j \in \{d, h\}}$  contain the normalized eigenvectors and the corresponding eigenvalues of  $\{\Sigma_j(s_i)\}_{j \in \{d, h\}}$ , respectively. We calculate the principal components (PC) by rotating the data along the direction of the eigenvectors (the direction of maximum variance) of the covariance matrix [41] as in the following:

$$\mathbf{P}_d(s_i) = \mathbf{D}'(s_i) \mathbf{U}_d(s_i) \quad (19)$$

$$\mathbf{P}_h(s_i) = \mathbf{H}'(s_i) \mathbf{U}_h(s_i). \quad (20)$$

Each eigenvalue  $\lambda_j(s_i)$  represents the amount of variance in the data, which is explained by the corresponding PC  $\mathbf{p}_j(s_i)$ . For instance, let us consider the daily performance patterns. The strongly correlated (pointing in the similar direction) error profiles  $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$  for link  $s_i$  imply that the prediction errors across different days  $\{j\}_{j=1}^{m_d}$  follow similar patterns. In this case, few PC  $f_d(s_i)$  can cover most of the variance in the daily error performance data  $\mathbf{D}'(s_i)$  of  $s_i$  [41]. If most days show independent behavior, then we would require more PC  $f'_d(s_i)$  to explain the same percentage of variance in the data.

The same goes for the hourly error patterns. For a link  $s_i$  with similar performance across different hours, we require a small number of hourly PC  $f_h(s_i)$ . In case of large hourly performance variations, we will need a large number of hourly PC to explain the same amount of variance.

The number of PC  $\{f_j(s_i)\}_{j \in \{d, h\}}$  is chosen using a certain threshold of total variance  $\eta_\sigma$  (typically 80%) in the data

[41]. We define the following distance measure to compare the consistency in the prediction performance of two links:

$$\Delta_t(s_i, s_j) = \sqrt{(f_d(s_i) - f_d(s_j))^2 + (f_h(s_i) - f_h(s_j))^2}. \quad (21)$$

We find the clusters of consistent  $\tau_1$  and inconsistent  $\tau_2$  links by applying (21) and the  $k$ -means clustering. The consistent (inconsistent) links will have similar (variable) performance patterns across days and during different hours.

2) *Link-Level Temporal Prediction Patterns*: In the previous section, we proposed a method to find the consistent and inconsistent links. In this section, we propose an algorithm to cluster the days/hours with similar performance for each road segment  $s_i$ . The algorithm also conserves the topological relation between the clusters. The topological relations are considered in the sense of the mean prediction performance of the different clusters [42]. To this end, we use the SOMs. The SOMs belong to the category of neural networks that can perform unsupervised clustering. In the SOM, each cluster is represented by a neuron. Neurons are organized in a grid pattern  $\mathfrak{M}$ . The weight  $\{\mathbf{q}_\rho\}_{\rho \in \mathfrak{M}}$  of the neuron represents the center of cluster  $\rho$ . We use the Kohonen rule [42] to find the optimal weights (cluster centers).

Consider a road segment  $s_i$  with prediction performance matrix  $\mathbf{D}(s_i)$ . Matrix  $\mathbf{D}(s_i)$  is composed of the daily prediction error profiles  $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$  for  $m_d$  days. Let us represent each day by index  $\{j\}_{j=1}^{m_d}$ . We aim to identify the subset (cluster) of the days  $\rho \subseteq \{j\}_{j=1}^{m_d}$  with similar performance patterns. Moreover, we aim to find a 2-D grid  $\mathfrak{M}$  for the clusters. In the grid, the clusters  $\rho \in \mathfrak{M}$  with similar behavior (daily prediction performance) will be placed adjacent to each other. However, each daily performance profile contains the data for multiple prediction horizons and time instances. It is hard to visualize the data in such high-dimensional representation. We apply the SOM to visualize and map the daily performance patterns on a 2-D clustering grid  $\mathfrak{M}$  [43].

We apply the same procedure to find different groups of hourly patterns with similar prediction performance for each road segment  $s_i$ . To this end, the SOM performs the clustering by considering the hourly profile matrix  $\mathbf{H}(s_i)$  for each link  $s_i$ .

In this section, we have proposed unsupervised learning methods to find the spatiotemporal performance patterns. In the following section, we apply these proposed performance analysis methods to the prediction data of the SVR and provide results.

## VII. RESULTS AND DISCUSSION

Let us start with the spatial performance patterns. We apply the  $k$ -means clustering to find the road segments with similar performance across the prediction horizons. We consider three different validation indexes to obtain the optimal number of clusters for the test network. All three validation methods yield four clusters as the optimal structure. The spatial distribution of the different clusters is shown in Fig. 5. The error distributions for each cluster across different prediction horizons are shown in Fig. 6.

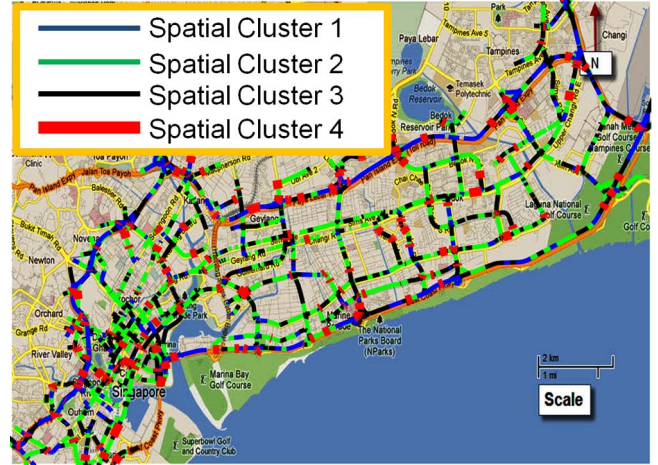


Fig. 5. Road segments belonging to the different spatial clusters.

The first cluster ( $\omega_1$ ) consists of roads with high prediction accuracy (see Table II). We refer to this group of links as the best performing cluster (Cluster 1 or  $\omega_1$ ). Most of the roads in the network [around 75%, see Fig. 7(b)] belong to Clusters 2 ( $\omega_2$ ) and 3 ( $\omega_3$ ). These two clusters represent the performance trends of the majority of roads in the network. It is interesting to note that the combined prediction performance of these two clusters (see Table II) is worse than the mean prediction accuracy of the whole network [see Fig. 1(a)]. In this case, the network-wise MAPE [see Fig. 1(a)] provides a slightly inflated depiction of the overall prediction accuracy. This is due to the high prediction accuracy of the best performing cluster. Finally, the roads that belong to Cluster 4 ( $\omega_4$ ) have the highest prediction error from each road category (see Table II). The proposed prediction algorithm performs poorly for this set of road segments. We refer to this cluster as the worst performing cluster.

The spatial clusters also have some intuitive sense. For instance, most of the expressways belong to the best performing cluster [around 80%, see Fig. 7(c)]. However, a small group of roads from the other categories also belong to this cluster. Interestingly, a small percentage of expressways also appear in other clusters [see Fig. 7(c)]. Some expressway sections even appear in Cluster 4 ( $\omega_4$ ). The expressway sections that belong to  $\omega_4$  are mostly situated at busy exits. Naturally, it is relatively hard to predict traffic on such sections. Moreover, a higher ratio of Category C (CATC) roads belong to  $\omega_3$  compared with the roads in Category B (CATB) [see Fig. 7(c)]. Likewise, the majority of the CATD and CATE roads [referred to as “others” in Fig. 7(c)] also belong to the spatial Cluster 3.

Overall, the roads within each spatial cluster show similar performance (see Fig. 6). Consequently, we find a small SD of error within each cluster across the prediction horizons [see Fig. 7(a)]. The behavior of the worst performing cluster is an exception in this case.

The spatial clusters can also provide useful information about the relative predictability of different road segments. Consider the intersection that is shown in Fig. 7(d). It shows that the roads carrying inbound traffic may have different prediction performance compared with the roads carrying outbound traffic. In

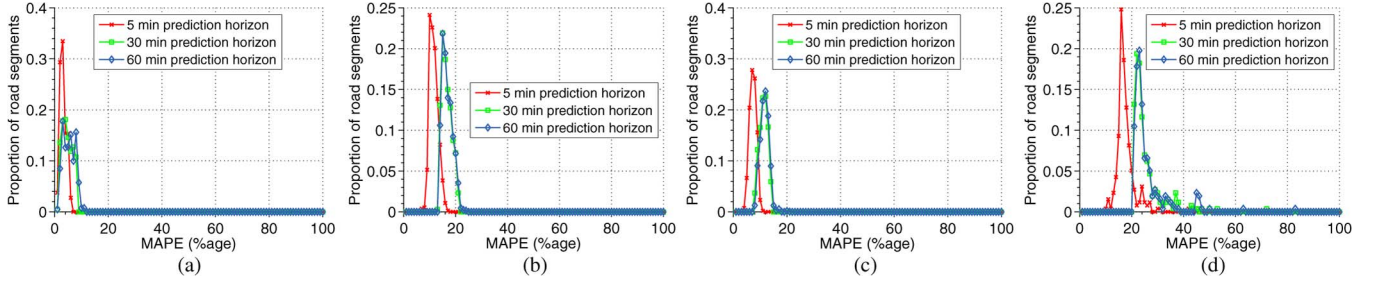
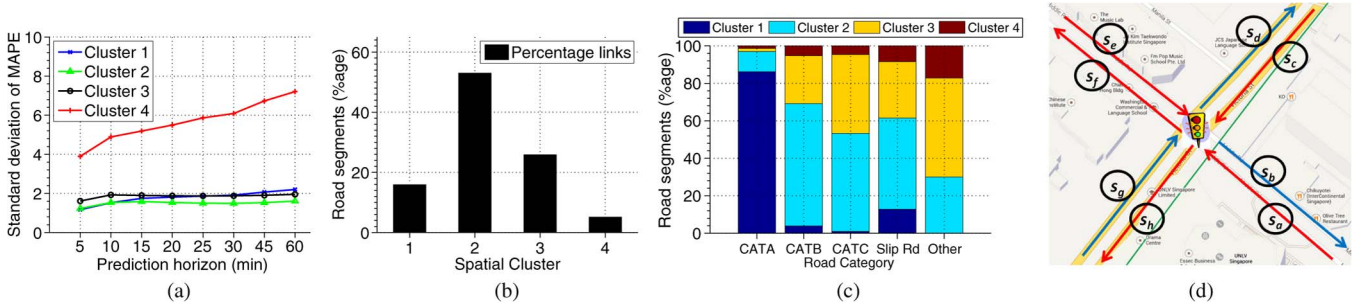


Fig. 6. Error distribution for the different clusters. (a) Cluster 1. (b) Cluster 2. (c) Cluster 3. (d) Cluster 4.

TABLE II  
PERFORMANCE OF THE DIFFERENT SPATIAL CLUSTERS

Cluster	Prediction Horizon							
	5 min	10 min	15 min	20 min	25 min	30 min	45 min	60 min
Cluster 1 ( $\omega_1$ )	2.69%	3.24%	3.66%	3.86%	4.05%	4.20%	4.61%	4.89%
Cluster 2 ( $\omega_2$ )	6.79%	9.16%	10.38%	10.54%	10.64%	10.72%	10.95%	11.14%
Cluster 3 ( $\omega_3$ )	11.06%	14.60%	15.90%	16.01%	16.05%	16.08%	16.18%	16.28%
Cluster 4 ( $\omega_4$ )	17.18%	23.04%	24.58%	24.81%	24.98%	25.10%	25.41%	25.62%

Fig. 7. Properties of the spatial clusters. In Fig. 7(d), the blue links ( $s_b, s_d, s_g$ ) and the red links ( $s_a, s_c, s_e, s_f, s_h$ ) correspond to spatial clusters  $\omega_2$  and  $\omega_3$ , respectively. (a) Intracluster error SD. (b) Percentage distribution of road segments. (c) Road categories and spatial clusters. (d) Sections belonging to the different clusters at a particular intersection.TABLE III  
DISTRIBUTION OF LINKS IN TEMPORAL CLUSTERS

Temporal Cluster ( $\tau$ )	Cluster Center		Spatial Cluster ( $\omega$ )				Total links
	$f_d$	$f_h$	cluster 1 ( $\omega_1$ )	cluster 2 ( $\omega_2$ )	cluster 3 ( $\omega_3$ )	cluster 4 ( $\omega_4$ )	
Cluster 1 ( $\tau_1$ )	5	6	735	610	125	30	1500
Cluster 2 ( $\tau_2$ )	13	12	66	2054	1176	228	3524

this case, the links carrying traffic toward the downtown area ( $s_a, s_c, s_f, s_h$ ) tend to show degraded prediction performance (Cluster 3).

We apply the PCA and the  $k$ -means clustering to find the temporal performance patterns. We create two clusters for the road segments in this regard. We refer to these clusters as the consistent and inconsistent clusters ( $\tau_1$  and  $\tau_2$ , respectively). Table III summarizes the properties of these two clusters. The prediction performance of road segments in the consistent cluster remains uniform across days and during different hours. The links in the inconsistent cluster have variable prediction performance during different time periods.

We observe that the roads with similar mean prediction performance [see Fig. 7(a)] can still have different temporal performance patterns [see Table III]. All these spatial

clusters (Clusters 1, 2, and 3) have a small intracluster SD [see Fig. 7(a)]. The majority of roads in the spatial Cluster 1 (the best performing cluster) are part of the consistent cluster. However, a small proportion of roads from the best performing cluster are also part of the inconsistent cluster. We observe this trend in the other spatial clusters as well. Although the majority of the links in the spatial Clusters 2 and 3 are part of the inconsistent cluster, they both still have a sizeable proportion of consistent links (see Table III). Even in the case of the worst performing cluster, a small subset of roads are part of the consistent cluster. These road segments consistently report high prediction error during most of the time periods.

The temporal performance analysis shows that the links with similar overall prediction behavior can still have variable temporal performance. To analyze such trends in detail, we

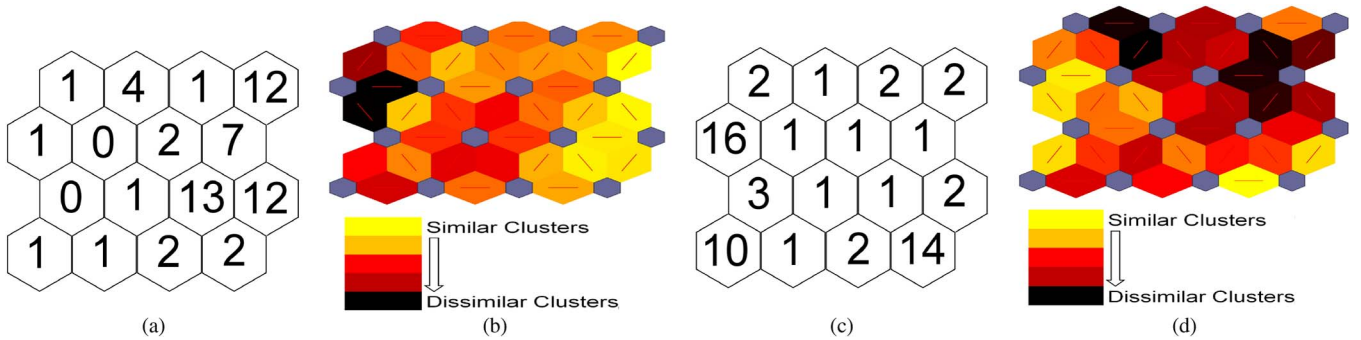


Fig. 8. Daily performance patterns for road segments  $s_1$  (the consistent link) and  $s_2$  (the inconsistent link). Hexagons represent the clusters, and the entry within each hexagon denotes the number of days in that cluster. The color bar represents the cluster similarity in terms of Euclidean distance. (a) Distribution of the different clusters (road segment  $s_1$ ). (b) Similarity between the different clusters (road segment  $s_1$ ). (c) Distribution of the different clusters (road segment  $s_2$ ). (d) Similarity between the different clusters (road segment  $s_2$ ).

focus on two specific links  $s_1$  and  $s_2$  in the network, which have the following properties: They both belong to the same road category (CATA), and both of them are from the best performing cluster. However, road segment  $s_1$  is part of the consistent cluster, and  $s_2$  is from the inconsistent cluster. We apply the SOM to analyze the variations in the daily performance patterns of these two links.

Fig. 8(a) and (b) shows the different properties of the daily performance patterns for the consistent link  $s_1$ . In Fig. 8(a), we present the composition of each cluster for  $s_1$ . The entry within each hexagon denotes the number of days that belong to that cluster. Fig. 8(b) shows the relative similarity of each cluster and its neighbors. For the consistent link, we find that most of the days fall into four clusters [see Fig. 8(a)]. The SOM helps us conserve the topological relations of these clusters. The four main clusters are positioned adjacent to each other [see Fig. 8(a)]. This implies that these clusters represent the days with similar daily performance patterns [see Fig. 8(b)]. For this road segment, we observe similar performance patterns for most of the days [see Fig. 8(b)].

Now, let us consider the behavior of the inconsistent link  $s_2$ . Fig. 8(c) and (d) shows the prediction patterns for the road segment. In this case, we find three major clusters. The rest of the days are scattered into other small clusters [see Fig. 8(c)]. Even these three clusters represent quite different daily performance patterns [see Fig. 8(d)].

Both of these links belong to the same road category and spatial cluster. However, their daily performance patterns are quite different from each other. In case of the consistent link, we observe that the prediction error patterns do not significantly vary on a daily basis. For the inconsistent link, the performance patterns significantly vary from one day to another.

ITS applications, such as route guidance, which rely on the prediction data, are vulnerable to variations in prediction error. The spatiotemporal performance patterns provide insight into the prediction behavior of different road segments. Consider the example of a route guidance application. The route guidance algorithm can assign large penalties to the links that belong to the clusters with large prediction errors (e.g., clusters  $\omega_3$  and  $\omega_4$ ). For instance, with the spatial clustering, we can see that planning routes by incorporating expressway sections from cluster  $\omega_4$  (the worst performing links) may not be a good

idea. The spatial clusters also serve another important purpose. They provide information about the relative predictability of the different road segments in a particular network. Furthermore, the temporal clusters can help the algorithm to avoid inconsistent links. These links might have a low average prediction error. However, their prediction performance may widely vary from one time instance to another. Again, consider the example of a route guidance algorithm. It would be better to plan a route by incorporating the roads with known performance patterns, although they have a slightly larger prediction error than the inconsistent roads. The application can utilize these spatiotemporal markers to provide routes that are more robust to variations in prediction performance.

## VIII. CONCLUSION

In this paper, we have proposed unsupervised learning methods to analyze the spatiotemporal performance trends in an SVR-based large-scale prediction. We have performed a large-scale traffic prediction for multiple prediction horizons with the SVR. Our analysis focused on a large and heterogeneous road network. The SVR produced better prediction accuracy in comparison with other forecasting algorithms. We have also observed that traditional performance evaluation indexes such as the MAPE fail to provide any insight about performance patterns. For instance, traffic speeds on some roads were found to be more predictable than others, and their performance remained uniform across different time periods. For some other roads, such patterns significantly varied across time. We have proposed unsupervised learning methods to infer these patterns in a large-scale prediction. We have used the prediction data from the SVR to assess the efficiency of these unsupervised learning algorithms. These methods provide a systematic approach for evaluating the predictability and performance consistency of different road segments. Such insights may be useful for the ITS applications that use prediction data to achieve time-sensitive objectives.

For future works, we propose to incorporate these performance patterns into predictive route guidance algorithms. This can lead to the development of route recommendation algorithms that are more robust to variations in the future state of road networks.

## REFERENCES

- [1] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.
- [2] J. Park, D. Li, Y. Murphey, J. Kristinsson, R. McGee, M. Kuang, and T. Phillips, "Real time vehicle speed prediction using a neural network traffic model," in *Proc. IJCNN*, 2011, pp. 2991–2996.
- [3] C. Wu, J. Ho, and D. Lee, "Travel-time prediction with support vector regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 276–281, Dec. 2004.
- [4] L. Vanajakshi and L. Rilett, "A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed," in *Proc. IEEE Intell. Veh. Symp.*, 2004, pp. 194–199.
- [5] Y. Zhang and Y. Liu, "Traffic forecasting using least squares support vector machines," *Transportmetrica*, vol. 5, no. 3, pp. 193–213, Sep. 2009.
- [6] C. Quek, M. Pasquier, and B. Lim, "POP-TRAFFIC: A novel fuzzy neural approach to road traffic analysis and prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 133–146, Jun. 2006.
- [7] L. Vanajakshi and L. Rilett, "Support vector machine technique for the short term prediction of travel time," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 600–605.
- [8] E. Vlahogianni, M. Karlaftis, and J. Golias, "Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach," *Transp. Res. C, Emerging Technol.*, vol. 13, no. 3, pp. 211–234, Jun. 2005.
- [9] A. Ding, X. Zhao, and L. Jiao, "Traffic flow time series prediction based on statistics learning theory," in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, 2002, pp. 727–730.
- [10] C. van Hinsbergen, J. Van Lint, and H. Van Zuylen, "Bayesian committee of neural networks to predict travel times with confidence intervals," *Transp. Res. C, Emerging Technol.*, vol. 17, no. 5, pp. 498–509, Oct. 2009.
- [11] D. Park and L. Rilett, "Forecasting freeway link travel times with a multilayer feedforward neural network," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 14, no. 5, pp. 357–367, Sep. 1999.
- [12] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 871–882, Jun. 2013.
- [13] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. C, Emerging Technol.*, vol. 19, no. 4, pp. 606–616, Aug. 2011.
- [14] B. Smith, B. Williams, and R. Keith Oswald, "Comparison of parametric and nonparametric models for traffic flow forecasting," *Transp. Res. C, Emerging Technol.*, vol. 10, no. 4, pp. 303–321, Aug. 2002.
- [15] Z. Sun, Y. Wang, and J. Pan, "Short-term traffic flow forecasting based on clustering and feature selection," in *Proc. IEEE IJCNN World Congr. Comput. Intell.*, 2008, pp. 577–583.
- [16] J. McFadden, W. Yang, and S. Durrans, "Application of artificial neural networks to predict speeds on two-lane rural highways," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1751, no. 1, pp. 9–17, 2001.
- [17] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, Nov. 2003.
- [18] M. Van Der Voort, M. Dougherty, and S. Watson, "Combining Kohonen maps with arima time series models to forecast traffic flow," *Transp. Res. C, Emerging Technol.*, vol. 4, no. 5, pp. 307–318, Oct. 1996.
- [19] P. Lingras and P. Mountford, "Time delay neural networks designed using genetic algorithms for short term inter-city traffic forecasting," in *Engineering of Intelligent Systems*. Berlin, Germany: Springer-Verlag, 2001, pp. 290–299.
- [20] X. Jiang and H. Adeli, "Dynamic wavelet neural network model for traffic flow forecasting," *J. Transp. Eng.*, vol. 131, no. 10, pp. 771–779, Oct. 2005.
- [21] J. Zhu and T. Zhang, "A layered neural network competitive algorithm for short-term traffic forecasting," in *Proc. Int. Conf. CISE*, 2009, pp. 1–4.
- [22] A. Smola and B. Schölkopf, "A tutorial on support vector regression," *Stat. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004.
- [23] C. Chang and C. Lin, "Training v-support vector regression: Theory and algorithms," *Neural Comput.*, vol. 14, no. 8, pp. 1959–1977, Aug. 2002.
- [24] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [25] S. Sundaram, H. N. Koutsopoulos, M. Ben-Akiva, C. Antoniou, and R. Balakrishna, "Simulation-based dynamic traffic assignment for short-term planning applications," *Simul. Model. Pract. Theory*, vol. 19, no. 1, pp. 450–462, Jan. 2011.
- [26] P. Moriarty and D. Honnery, "Low-mobility: The future of transport," *Futures*, vol. 40, no. 10, pp. 865–872, Dec. 2008.
- [27] K. Müller, A. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik, "Predicting time series with support vector machines," in *Proc. ICANN*, 1997, pp. 999–1004.
- [28] B. Schölkopf, P. Bartlett, A. Smola, and R. Williamson, "Shrinking the tube: a new support vector regression algorithm," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 1999, pp. 330–336.
- [29] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, May 2011.
- [30] S. Keerthi and C. Lin, "Asymptotic behaviors of support vector machines with Gaussian kernel," *Neural Comput.*, vol. 15, no. 7, pp. 1667–1689, Jul. 2003.
- [31] R. Frank, N. Davey, and S. Hunt, "Time series prediction and neural networks," *J. Intell. Robot. Syst.*, vol. 31, no. 1–3, pp. 91–103, May 2001.
- [32] H. Chen and S. Grant-Muller, "Use of sequential learning for short-term traffic flow forecasting," *Transp. Res. C, Emerging Technol.*, vol. 9, no. 5, pp. 319–336, Oct. 2001.
- [33] B. Williams, P. Durvasula, and D. Brown, "Urban freeway traffic flow prediction: Application of seasonal autoregressive integrated moving average and exponential smoothing models," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1644, no. 1, pp. 132–141, 1998.
- [34] M. Gori and A. Tesi, "On the problem of local minima in back propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 1, pp. 76–86, Jan. 1992.
- [35] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.
- [36] K. Wang, B. Wang, and L. Peng, "CVAP: Validation for cluster analyses," *Data Sci. J.*, vol. 8, no. 20, pp. 88–93, 2009.
- [37] R. Sharan, A. Maron-Katz, and R. Shamir, "Click and expander: A system for clustering and visualizing gene expression data," *Bioinformatics*, vol. 19, no. 14, pp. 1787–1799, Sep. 2003.
- [38] P. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.
- [39] J. Hartigan, *Clustering Algorithms*. Hoboken, NJ, USA: Wiley, 1975.
- [40] I. Pelczar and H. Cisneros-Iturbe, "Identification of rainfall patterns over the valley of Mexico," in *Proc. 11th Int. Conf. Urban Drainage*, Edinburgh, U.K., 2008, pp. 1–9.
- [41] I. Jolliffe, *Principal Component Analysis*. Hoboken, NJ, USA: Wiley, 2005.
- [42] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, Sep. 1990.
- [43] B. S. Penn, "Using self-organizing maps to visualize high-dimensional data," *Comput. Geosci.*, vol. 31, no. 5, pp. 531–544, Jun. 2005.



**Muhammad Tayyab Asif** (S'12) received the Bachelor of Engineering degree from the University of Engineering and Technology Lahore, Lahore, Pakistan. He is currently working toward the Ph.D. degree in the School of Electrical and Electronic Engineering, College of Engineering, Nanyang Technological University, Singapore.

Previously, he was with Ericsson as a Design Engineer in the domain of mobile packet core networks. His research interests include sensor fusion, network optimization, and modeling of large-scale networks.



**Justin Dauwels** (M'09–SM'12) received the Ph.D. degree in electrical engineering from the Swiss Federal Institute of Technology in Zurich (ETHZ), Zurich, Switzerland, in December 2005.

He is currently an Assistant Professor with the School of Electrical and Electronic Engineering, College of Engineering, Nanyang Technological University, Singapore. His research interests include Bayesian statistics, iterative signal processing, and computational neuroscience.

Dr. Dauwels has been a Japan Society for the Promotion of Science (JSPS) Postdoctoral Fellow in 2007, a Belgian American Education Foundation fellow and a Henri Benedictus Fellow of the King Baudouin Foundation in 2008, and a JSPS invited fellow in 2010.



**Chong Yang Goh** received the B.Eng. degree from Nanyang Technological University, Singapore. He is currently working toward the Ph.D. degree at the Massachusetts Institute of Technology, Cambridge, MA, USA.

His research interests include the development of real-time decision models and algorithms that incorporate large-scale asynchronous data sources, with applications in transportation.



**Menoth Mohan Dhanya** received the Bachelor's degree in mechatronics engineering from Anna University of Technology, Chennai, India, and the Master's degree in computer control and automation from Nanyang Technological University, Singapore.

She is currently a Research Associate with the School of Electrical and Electronic Engineering, College of Engineering, Nanyang Technological University, Singapore. Her research interests include machine learning, signal processing, and automation.



**Ali Oran** received the Master's and Doctoral degrees from the University of California San Diego, CA, USA, in 2005 and 2010, respectively.

He is currently a Postdoctoral Associate with the Future Urban Mobility Group, Singapore–Massachusetts Institute of Technology Alliance for Research Technology (SMART) Center, Singapore. His research interests include vehicle localization, vehicle routing, and decision making under uncertainty.



**Nikola Mitrovic** received the Bachelor's degree in traffic engineering from the University of Belgrade, Belgrade, Serbia, in 2009 and the Master's degree from Florida Atlantic University, Boca Raton, FL, USA, in 2010. He is currently working toward the Ph.D. degree in the School of Electrical and Electronic Engineering, College of Engineering, Nanyang Technological University, Singapore.

His research interests include traffic modeling, intelligent transportation systems, and transportation planning.



**Esmail Fathi** was born in Loshan, Iran. He received the Master's degree from Shahid Beheshti University, Tehran, Iran, in 2007.

He was an Associate Researcher with Nanyang Technological University, Singapore, researching on the topic of tracking and predicting traffic flow in dynamic urban transportation networks. He is currently working as a Project Engineer with Siemens Private Ltd., Singapore.



**Patrick Jaillet** received the Ph.D. degree in operations research from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1985.

He is currently the Dugald C. Jackson Professor of the Department of Electrical Engineering and Computer Science, School of Engineering, and a Codirector of the Operations Research Center, Massachusetts Institute of Technology. His research interests include algorithm design and analysis for online problems, real-time and dynamic optimization, network design and optimization, and prob-

abilistic combinatorial optimization.



**Muye Xu** received the B.Eng. (Honors) degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2012.

He is currently a Technology Analyst with the Trading Technology Group, Deutsche Bank AG, Singapore. His research interest includes machine learning, transportation systems, and operations research.