WILEY | Hindawi

Research Article

Network Traffic Prediction Based on Deep Belief Network and Spatiotemporal Compressive Sensing in Wireless Mesh Backbone Networks

Laisen Nie,[1] Xiaojie Wang (iD),[2] Liangtian Wan (iD),[2] Shui Yu,[3] Houbing Song,[4] and Dingde Jiang[5]

[1]*School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China*
[2]*School of Software, Dalian University of Technology, Dalian 116620, China*
[3]*School of Information Technology, Deakin University, Burwood, VIC 3125, Australia*
[4]*Department of Electrical, Computer, Software, and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA*
[5]*School of Astronautics & Aeronautics, University of Electronic Science and Technology of China, Chengdu 611731, China*

Correspondence should be addressed to Xiaojie Wang; wangxj1988@mail.dlut.edu.cn

Wireless mesh network is prevalent for providing a decentralized access for users and other intelligent devices. Meanwhile, it can be employed as the infrastructure of the last few miles connectivity for various network applications, for example, Internet of Things (IoT) and mobile networks. For a wireless mesh backbone network, it has obtained extensive attention because of its large capacity and low cost. Network traffic prediction is important for network planning and routing configurations that are implemented to improve the quality of service for users. This paper proposes a network traffic prediction method based on a deep learning architecture and the Spatiotemporal Compressive Sensing method. The proposed method first adopts discrete wavelet transform to extract the low-pass component of network traffic that describes the long-range dependence of itself. Then, a prediction model is built by learning a deep architecture based on the deep belief network from the extracted low-pass component. Otherwise, for the remaining high-pass component that expresses the gusty and irregular fluctuations of network traffic, the Spatiotemporal Compressive Sensing method is adopted to predict it. Based on the predictors of two components, we can obtain a predictor of network traffic. From the simulation, the proposed prediction method outperforms three existing methods.

1. Introduction

The Wireless Mesh Network (WMN) provides ubiquitous and last few miles connectivity for future wireless service, for example, IoT, 5G mobile network, and cognitive radio. Besides, it is also a promising solution to IoT crowdsensing applications by connecting a large group of individuals with capacities of computing and sensing. Compared with other wireless architectures (e.g., ad hoc networks), the WMN has high capacity, robustness, and low-cost deployment [1]. Hence, it is much more popular as an emerging access paradigm in practice. With the rapid development of mobile communications, mobile cloud systems, and IoT, the applications provided by wireless networks become multitudinous.

Besides, the big data has become a crucial role in both industry and daily life. The quantity and substantial growth of WMN (both in scale and in service) bring a series of new challenges for the capacity; for example, the network congestion may appear in a wireless mesh backbone network induced by huge traffic demands.

Mesh routers constitute the principal infrastructure of the WMN known as the wireless mesh backbone network. The self-organized manner of the WMN architecture reinforces the resilience of the network to failures, but it arises some limitations, typically, resource allocation problem, and so on. Above all, imperative network management operation is useful to provide a cost-effective solution for improving
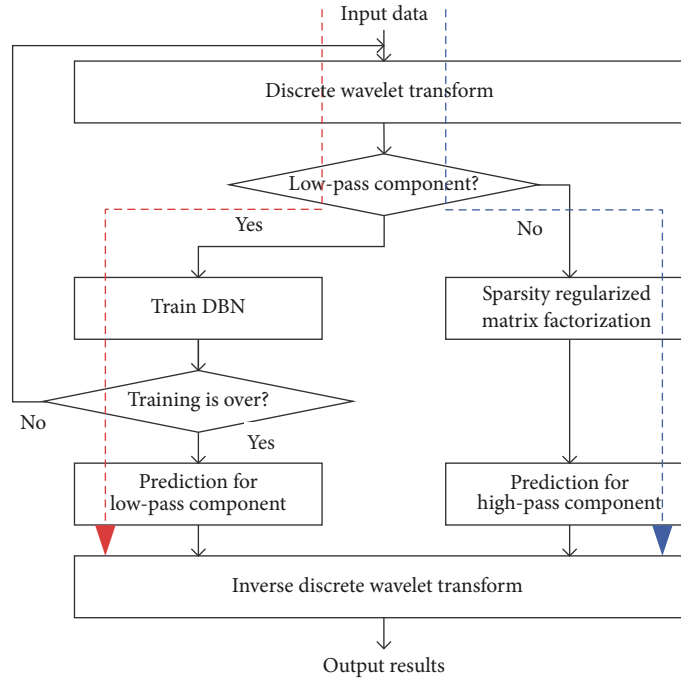
FIGURE 1: Prediction method.

the performance of a WMN. These network management operations are implemented in terms of the relative network traffic information. For instance, in order to improve the quality of service for users, predictive network planning is necessary for ISPs. This planning is carried out according to the future traces of network traffic flows between all possible origin-destination (OD) node pairs [2, 3].

A great number of methods have been proposed to deal with the network traffic prediction problem in traditional IP backbone networks [4–8]. Statistical methods have been widely adopted in this field. Originally, some simple models, such as Autoregressive (AR) and Autoregressive Integrated Moving Average (ARIMA), are used to pursue the short-range dependence (SRD) of network traffic [4]. However, current network traffic exhibits a long-range dependence (LRD) characteristic and multifractal features in terms of the behaviors of terminals [4]. In this case, the Fractional Autoregressive Integrated Moving Average (FARIMA) model and the Multifractal Wavelet model (MWM) are involved in this field to deal with the network traffic prediction problem [5]. With the variety of network services and applications, the characteristics of network traffic are much more complex. For instance, it exhibits some nonlinear features [4, 6]. Hence, some methods refer to the GARCH model to model network traffic for prediction. Besides, a lot of methods based on hybrid models have been proposed to predict network traffic [7]. However, these methods are not suitable for dealing with the problems of network traffic prediction in a wireless mesh backbone network [2]. Generally, the users join in a Wireless Mesh Network randomly. Additionally, they often have the complicated individual association, which is significantly discrepant compared with the users of a traditional IP backbone network. Although the users' applications of two

networks are probably coincident, the dominant applications usually are distinguishing.

Motivated by this issue, we propose a network traffic prediction method based on the deep belief network (DBN) and the Spatiotemporal Compressive Sensing (STCS) method, named Deep Belief Network and Spatiotemporal Compressive Sensing (DBNSTCS) models. To the best of our knowledge, this is the first paper that focuses on the problem of traffic prediction for wireless mesh backbone networks and takes into account the spatiotemporal characteristic for prediction. We take account of the long-range dependence and irregular fluctuation behaviors of network traffic independently; see Figure 1 [9]. By the discrete wavelet transform (DWT), the network traffic is divided into two components tagged by scaling and discrete wavelet transform coefficients. Namely, the DWT just likes a filter that decomposes the network traffic into a low-pass component and a high-pass component. The low-pass component expresses the long-range dependence of network traffic, and the high-pass one declares the gusty and irregular fluctuations. The LRD means that the network traffic at any time depends upon multiple previous traffic data. The LRD of network traffic derives from a series of interacted factors (e.g., the behaviors of users). To describe these multifarious relationships, the former is predicted by a deep architecture based on DBN. The proposed architecture can deeply learn the LRD of network traffic. For the short-range and irregular fluctuations, the STCS method known as an excellent interpolation algorithm is employed to predict them.

The contributions of the paper are proposed as follows:

(i) We use the DWT to extract the low-pass and high-pass components of network traffic. The DWT of a

time series can be viewed as making this time series pass a low-pass filter and a high-pass filter, respectively. Hence, we can obtain the low-pass and high-pass components of network traffic. They show the low-pass approximation and the details of network traffic, respectively. In our method, we predict two types of coefficients independently.

(ii) We propose a deep architecture based on DBN to capture the low-pass component of network traffic. Under this architecture, by learning the built deep architecture in terms of a training set via known network traffic, the deep architecture can describe the LRD characteristic of traffic flows and carry out a prediction for network traffic.

(iii) We adopt the Spatiotemporal Compressive Sensing to fit the gusty and irregular fluctuations of network traffic. We first assume that the high-pass component obeys a spatiotemporal dependence. Then, we achieve a predictor of network traffic by the Sparsity Regularized Matrix Factorization (SRMF) method.

The remaining parts of this paper are organized as follows. Section 2 reviews the related work about network traffic prediction problem. In Section 3, we introduce some definitions about network traffic, the DWT techniques, the DBN theory, and the Spatiotemporal Compressive Sensing method, respectively. We propose our prediction method in Section 4. Then, we verify the performance of our method in Section 5. Section 6 concludes our work of this paper.

2. Related Work

Lots of researchers have investigated network traffic prediction that is instructive for congestion control, predictive network planning, and intelligent routing [10–14]. The existing network traffic prediction techniques consist of four categories: linear time series methods, nonlinear time series methods, hybrid model methods, and decomposition model methods.

The linear time series methods (e.g., AR, MA, and ARMA) are frequently used to model end-to-end traffic flows for prediction. According to novel research findings, the traffic flows exhibit observably nonlinear features under complex network users behaviors and various applications. Typically, the GARCH model in [10] is used to model the burst characteristics. Besides, neural network is also a valid method to track the traffic flows with nonlinear characteristic [12].

With the rapidly development of network services, the ISP network has been a heterogeneous and complex network. Traffic flows show manifold statistic characteristics such as LRD, SRD, heavy-tailed distribution, and multifractal feature. Therefore, researchers adopt some hybrid models to model the traffic flows with complicated distributions. The methods based on hybrid model take advantage of two or more models to capture the traces of traffic flows. In [4], the authors combine the ARIMA model with the GARCH model to fit several characteristics of traffic flows (i.e., LRD and SRD

characteristics). Meanwhile, the proposed hybrid model can also model the self-similarity and multifractal features of traffic flows. The autocorrelation function and the partial autocorrelation function are employed by the authors to determine the parameters in the hybrid model. The fourth method as mentioned in the above part is the decomposition model methods in which the traffic flows are divided into several components. Based on this decomposition, the gained components are, respectively, modeled and predicted. These methods can be viewed as an evolution of the hybrid model methods. In [13], the authors jointly use the Stationary Wavelet Transform (SWT), the Quantum Genetic Algorithm (QGA), and the Backpropagation Neural Network (BPNN) to implement traffic prediction of wireless network traffic. They first decompose the traffic flows by the SWT such that the traffic flows are made up of several stationary components. After that, all these components are predicted by a trained BPNN using the QGA. The authors in [14] decompose the traffic of a large scale cellular network into regular and random components by a classic time series decomposition method.

3. Background

3.1. Traffic Matrix. A traffic matrix is an expression form of network traffic. After collecting network traffic information, the operators implement appropriate network management functions in terms of this network traffic information. During this period, the network traffic information is expressed as the traffic matrix. If we denote an OD flow by $x_{p,q}(t)$ which describes the mean of the volume of traffic flow from the origin node p to the destination node q in the tth time slot, then the traffic matrix is defined as

$$X = \begin{bmatrix} x_{1,1}(1) & x_{1,1}(2) & \cdots & x_{1,1}(T) \\ x_{1,2}(1) & x_{1,2}(2) & \cdots & x_{1,2}(T) \\ \vdots & \vdots & \cdots & \vdots \\ x_{N,N}(1) & x_{N,N}(2) & \cdots & x_{N,N}(T) \end{bmatrix}, \quad (1)$$

where $p, q \in \{1, 2, \ldots, N\}$, and $t \in \{1, 2, \ldots, T\}$. This traffic matrix reports the network traffic data with T time slots. Generally, the length of time slot is 5 or 15 minutes.

3.2. Discrete Wavelet Transform. For a time series $f(t)$, it can be expressed by

$$f(t) = \sum_{n=-\infty}^{+\infty} c_{L,n} 2^{-L/2} \phi\left(\frac{t}{2^L} - n\right) + \sum_{l=1}^{L} \sum_{n=-\infty}^{+\infty} d_{l,n} 2^{-l/2} \psi\left(\frac{t}{2^l} - n\right), \quad (2)$$

where $c_{L,n}$ and $d_{l,n}$ are scaling and discrete wavelet transform coefficients. $2^{-L/2}\phi(t/2^L - n)$ is the scaling function at scale L. $2^{-l/2}\psi(t/2^l - n)$ is so-called the wavelet function. The scaling coefficients express a coarse approximation of $f(t)$. The
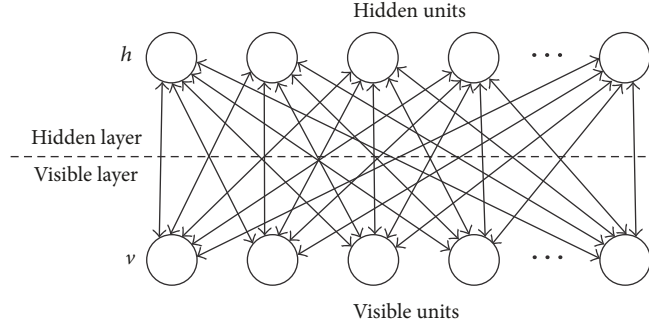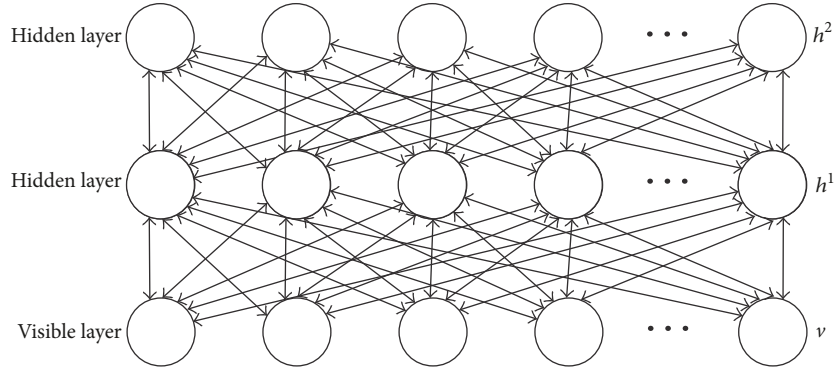
FIGURE 2: Restricted Boltzmann Machine.

FIGURE 3: Deep Belief Network with two Restricted Boltzmann Machines.

discrete wavelet transform coefficients represent the details of $f(t)$. Hence, (2) can be viewed as making the time series pass a filter and then obtaining a representation of $f(t)$ by the combination of low-pass and high-pass approximations.

3.3. Deep Belief Network.
The DBN is a common deep learning primitive. It is a combination of a number of Restricted Boltzmann Machines (RBMs) [15–17]. A RBM that is two-layer undirected graphical model consists of the visible and hidden layers denoted by v and h (shown by Figure 2) [9, 15]. Each unit in a layer is connected with all units of the other layer by undirected edges. The units in the same layer are disconnected with each other. Figure 3 shows an example of DBN architecture with two RBMs [9]. The DBN is a stack of many RBMs. The values of all units are stochastic variables [16]. Generally, they obey a Bernoulli distribution or a Gaussian distribution. When the visible and hidden units are Gaussian and Bernoulli, we have the following conditional distribution:

$$P\left(v_i \mid h\right) = N\left(b_i + \sum_{j=1}^{J} w_{i,j} h_j, 1\right),$$

$$P\left(h_j = 1 \mid v\right) = \mathrm{sigm}\left(a_j + \sum_{i=1}^{I} w_{i,j} v_i\right), \tag{3}$$

where $N(b_i + \sum_{j=1}^{J} w_{i,j} h_j, 1)$ denotes the Gaussian distribution whose mean and variance are $b_i + \sum_{j=1}^{J} w_{i,j} h_j$ and 1. $\mathrm{sigm}(z) =$

$\exp(z)/(1 + \exp(z))$ is the sigmoid function. I and J are the numbers of visible and hidden units, respectively [18]. b_i and a_j are the biases of visible and hidden units. $w_{i,j}$ expresses the symmetric interaction term between the visible unit v_i and the hidden unit h_j. For a RBM, the joint probability distribution function over visible and hidden units can be denoted by

$$P\left(v, h\right) = \frac{\exp\left(-E\left(v, h\right)\right)}{\sum_{v,h} \exp\left(-E\left(v, h\right)\right)}, \tag{4}$$

where $E(v, h)$ is termed as the energy function defined as

$$E\left(v, h\right) = -\frac{1}{2} \sum_{i=1}^{I} \left(b_i - v_i\right)^2 - \sum_{j=1}^{J} a_j h_j - \sum_{i=1}^{I} \sum_{j=1}^{J} w_{i,j} v_i h_j. \tag{5}$$

To train the DBN, the idea is to employ a layer-wise greedy strategy. Besides, the parameters are updated by minimizing the log probability $\log P(v)$.

3.4. Spatiotemporal Compressive Sensing and Sparsity Regularized Matrix Factorization.
Compressive sensing is a novel sampling technique for signal processing in recent years, which makes good use of the structure or redundancy of real-world signals. It takes advantage of an adaptive sampling scheme to sense these structural signals. In detail, the structure of these signals means that it can be denoted by a vector that just has several nonzero elements (i.e., the vector is sparse). In the adaptive sampling scheme, a random

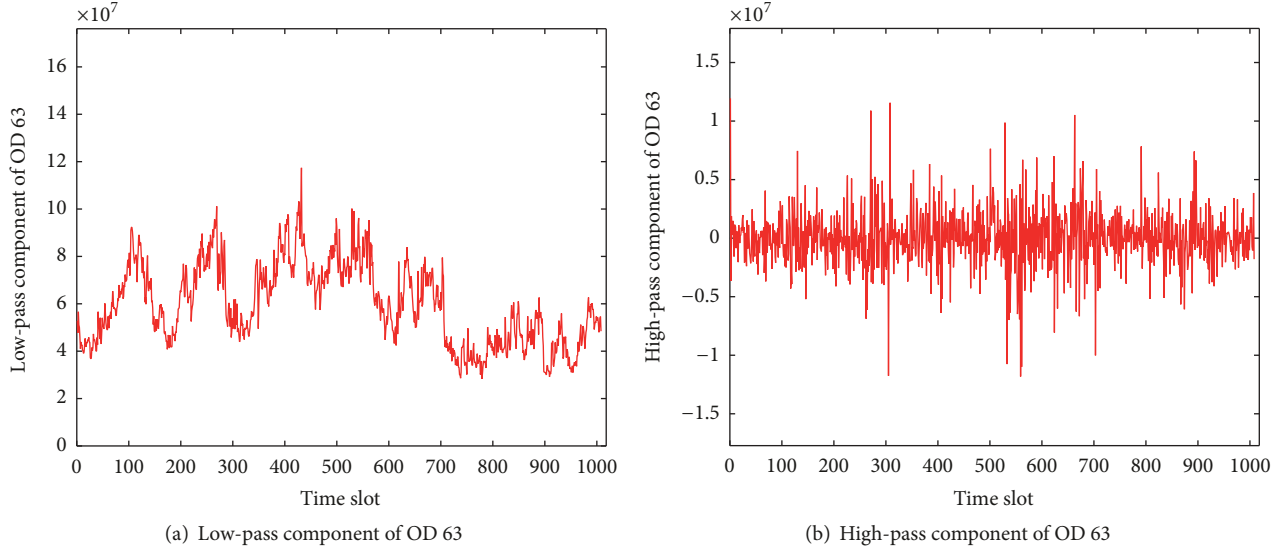(a) Low-pass component of OD 63

(b) High-pass component of OD 63

FIGURE 4: Decomposition of OD 63.

matrix called the measurement matrix is used to concurrently implement compressing and coding. During the decoding phase, the compressive sensing reconstruction algorithm is a splendid approach to deal with the inverse problem with ill-posed feature.

As a derivative of compressive sensing, the spatiotemporal compressive sensing technique is always used as an interpolation algorithm to recover the missing elements of a data set. The spatiotemporal feature means that the values of neighboring elements in the data set are properly similar. In terms of this feature, the SRMF method is proposed in [19], where the missing elements can be recovered precisely though the data loss probability is tremendous. Besides, the SRMF method is also an accurate tool for prediction. Under the prediction process, the elements that need to be predicted are viewed as continuous missing elements.

4. Our Methodology

4.1. Decomposition of Network Traffic.
We assume that the known network traffic is denoted by X whose each OD flow is denoted by a time series $x_{p,q}(t)$, where $t = 1, 2, \ldots, T$. According to (1), it can be denoted by

$$x_{p,q}(t) = \sum_{n=-\infty}^{+\infty} c_{L,n,p,q} 2^{-L/2} \phi\left(\frac{t}{2^L} - n\right)$$

$$+ \sum_{l=1}^{L} \sum_{n=-\infty}^{+\infty} d_{l,n,p,q} 2^{-l/2} \psi\left(\frac{t}{2^l} - n\right). \tag{6}$$

If we set the scale to be 1, then we have

$$x_{p,q}(t) = \sum_{n=-\infty}^{+\infty} c_{1,n,p,q} 2^{-1/2} \phi\left(\frac{t}{2} - n\right)$$

$$+ \sum_{n=-\infty}^{+\infty} d_{1,n,p,q} 2^{-1/2} \psi\left(\frac{t}{2} - n\right). \tag{7}$$

The above equation divides the network traffic into two components. One is the low-pass approximation (shown by the scaling coefficients) that exhibits the LRD of the network traffic $x_{p,q}(t)$, and the other is the high-pass approximation (described by the discrete wavelet transform coefficients) that expresses the gusty and irregular fluctuation behaviors of the network traffic $x_{p,q}(t)$. For a traffic matrix that describes the volume of traffic between all OD node pairs, obviously, its low-pass and high-pass approximation components can be denoted by two matrices, respectively.

Figures 4 and 5 give two examples of the traffic flow decomposition by the DWT. We select two OD flows from the real network traffic data set randomly, and plot their low-pass and high-pass components, respectively. Obviously, we see that the low-pass components are periodical, which means that they are much easier to be predicted comparing with the high-pass components shown by Figures 4(b) and 5(b). In this case, two components are predicted independently in this paper.

4.2. Deep Architecture for Low-Pass Component Prediction.
For an OD flow $x_{p,q}(t)$, we assume that the length of this series is an even number. In this case, the number of scaling coefficients is $T/2$. The deep architecture for prediction is plotted in Figure 6. There are M hidden layers in this architecture. Both the hidden and the input layers have $T/2$ units. At the top of the deep architecture, a single neuron defined as the logistic regression is employed for prediction. The logistic regression model is made up of a hidden layer with $T/2$ units and an output layer with one unit. The deep architecture is trained by the backpropagation algorithm in our method. The parameters of the deep architecture are determined lay by layer [18].

In our method, we first collect K training set denoted by $(x_{p,q}^1(t), \ldots, x_{p,q}^K(t))$. Then, we have the scaling coefficient training set $(c_{p,q}^1, \ldots, c_{p,q}^K)$. Meanwhile, the corresponding

(a) Low-pass component of OD 105
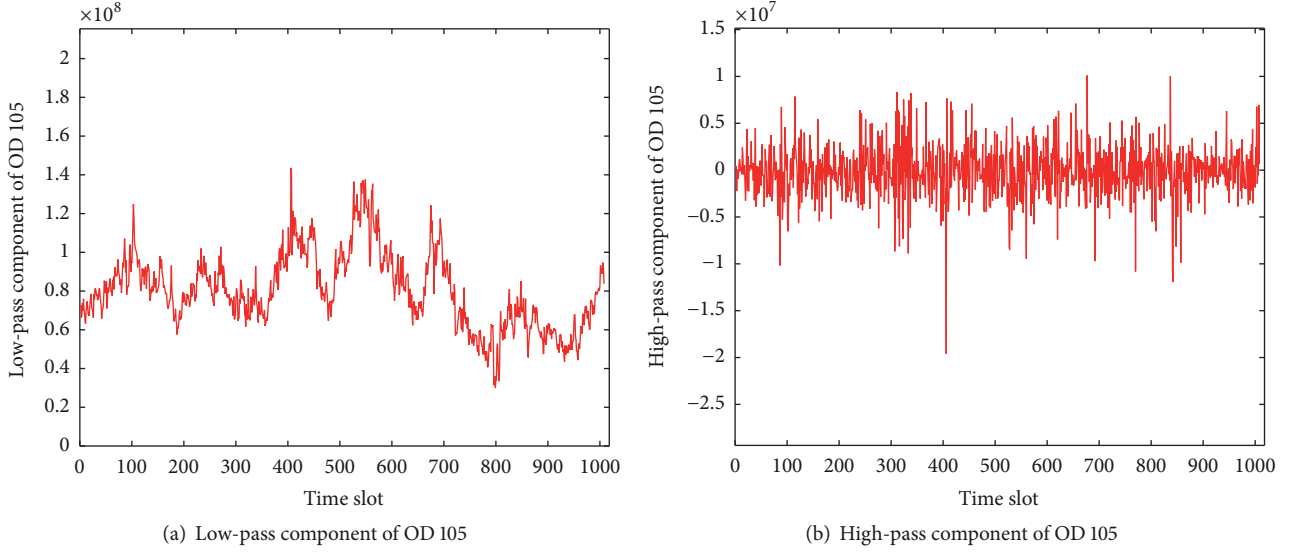

(b) High-pass component of OD 105
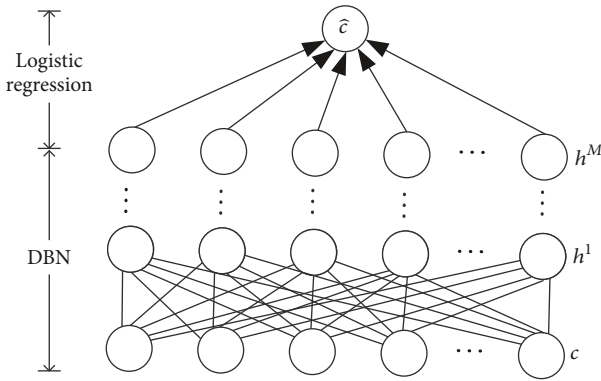
FIGURE 5: Decomposition of OD 105.

FIGURE 6: Deep architecture for prediction.

predictors are $(\hat{c}_{p,q}^1, \ldots, \hat{c}_{p,q}^K)$. By training the proposed deep architecture using the scaling coefficient training set, we can obtain a relationship between input and output scaling coefficients. Furthermore, we use the scaling coefficients of $x_{p,q}(t)$ as an input, and then a predictor of the scaling coefficient will be achieved.

4.3. Spatiotemporal Compressive Sensing for High-Pass Component Prediction. For the discrete wavelet transform coefficients, SRMF is a matrix-oriented interpolation algorithm. Therefore, different from the prediction of the scaling coefficients where each OD flow is predicted independently, we predict the discrete wavelet transform coefficients of all OD flows at the same time. According to (7), the discrete wavelet transform coefficients of all OD flows constitute a matrix denoted by D in this paper. The matrix D consists of two portions. One is from the training data obtained by measured network traffic data, and the other needs to be predicted. We denote the final predicting result by $\widehat{D}$, and then it

can be predicted by the following regularized optimization model:

$$\left\|\widehat{D} - D\right\|_F^2 + \lambda \left(\|L\|_F^2 + \|R\|_F^2 \right) + \left\|\left(LR^T\right)H\right\|_F^2, \qquad (8)$$

where the notations $\|\cdot\|_F$ and $(\cdot)^T$ denote the Frobenius norm and transposition, respectively. The matrix H, so-called the temporal constraint matrix (shown by (9)), describes the temporal neighbors.

$$H = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \qquad (9)$$

The matrices L and R are from the singular value decomposition of the matrix $\widehat{D}$; that is,

$$\widehat{D} = U\Sigma V^T. \qquad (10)$$

U and T are two unitary matrices, and Σ is a diagonal matrix whose diagonal elements are the singular values of $\widehat{D}$. The matrices L and R are equal to $U\Sigma^{1/2}$ and $V\Sigma^{1/2}$.

Finally, according to the predictors of the scaling and discrete wavelet transform coefficients, we predict the network traffic by inverse discrete wavelet transform. Algorithm 1 proposes the details of our method.

5. Simulation Results and Analysis

This section will verify the performance of our prediction method. In our simulations, a real network traffic data set with 2016 time slots is sampled on a time scale of 5 minutes. For exact prediction, the first 2000 time slots are used as the prior information to train the deep architecture and the

(1) **Input:** training set $(x^1_{p,q}(t), x^2_{p,q}(t), \ldots, x^K_{p,q}(t))$; known network traffic X
(2) **Output:** network traffic predictor $X(T + 1)$
(3) **for** $i = 1 : K$ **do**
(4) $c^i_{p,q} \leftarrow \text{DWT}(x^i_{p,q}(t))$
(5) **end for**
(6) Training deep architecture using $(c^1_{p,q}, c^2_{p,q}, \ldots, c^K_{p,q})$ as a training set
(7) $[C, D] \leftarrow \text{DWT}(X)$
(8) $\widehat{C} \leftarrow \text{DBN}(C)$
(9) $\widehat{D} \leftarrow \text{SRMF}(D)$
(10) $X(T + 1) \leftarrow \text{IDWT}(\widehat{C}, \widehat{D})$
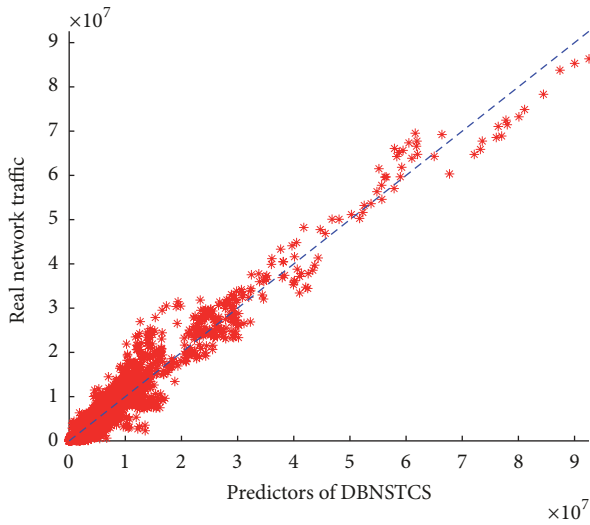(11) **return** $X(T + 1)$

ALGORITHM 1: DBNSTCS algorithm.

FIGURE 7: Real traffic data versus their predictors via DBNSTCS.
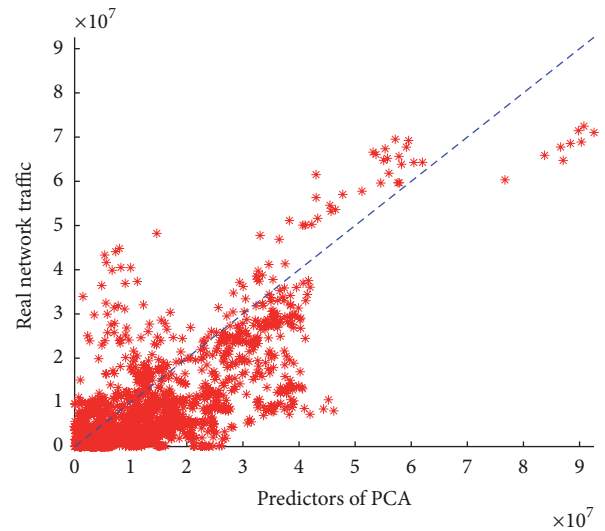

FIGURE 8: Real traffic data versus their predictors via PCA.

matrix D. We will compare our method with three state-of-the-art methods in network traffic prediction field, that is, the principal component analysis (PCA) method [20], the Tomogravity method [21], and the SRMF method [19]. The proposed method is implemented by MATLAB in a single machine with Core i5 central processing unit, 4 GB memory, and 1664 MB graphics processing unit (GPU) memory. Meanwhile, we set $M = 8$ and $T = 800$.

We first plot the real network traffic versus their predictors from four methods, respectively. Figure 7 displays the prediction results of our method. The x-axis and y-axis denote predictors and real network traffic, respectively. From Figure 7, we see that our method has low prediction biases for small network traffic flows. By contrast, our method shows positive predictions for large network traffic. The same conclusion can be obtained from Figure 8. For large network traffic, it also has positive predictions. For small network traffic, PCA has much larger prediction bias. Tomogravity has consistently positive predictions for large network traffic shown by Figure 9. For small network traffic, Tomogravity shows a desired prediction error. Besides, for large network

traffic, SRMF in Figure 10 has positive or negative predictions more or less.

Now, we refer to the spatial and temporal relative errors as a metric to compare four methods. The spatial and temporal relative errors are defined as

$$\text{SRE}(n) = \frac{\sqrt{\sum_{t=1}^{T}\left(\widehat{x}_n(t) - x_n(t)\right)^2}}{\sqrt{\sum_{t=1}^{T} x_n^2(t)}},$$

$$\text{TRE}(t) = \frac{\sqrt{\sum_{n=1}^{N^2}\left(\widehat{x}_n(t) - x_n(t)\right)^2}}{\sqrt{\sum_{n=1}^{N^2} x_n^2(t)}}, \tag{11}$$

where $x_n(t)$ and $\widehat{x}_n(t)$ are the nth end-to-end network traffic flow and its predictor. As mentioned above, $p, q \in \{1, 2, \ldots, N\}$; thus the number of OD flows is N^2. Figure 11(a) exhibits the spatial relative errors (SREs) of four methods. The x-axis is the identities of end-to-end network traffic flows. The end-to-end network traffic flows are sorted in descending order with respect to their means. Meanwhile, the y-axis is the SRE. From this simulation, we find that
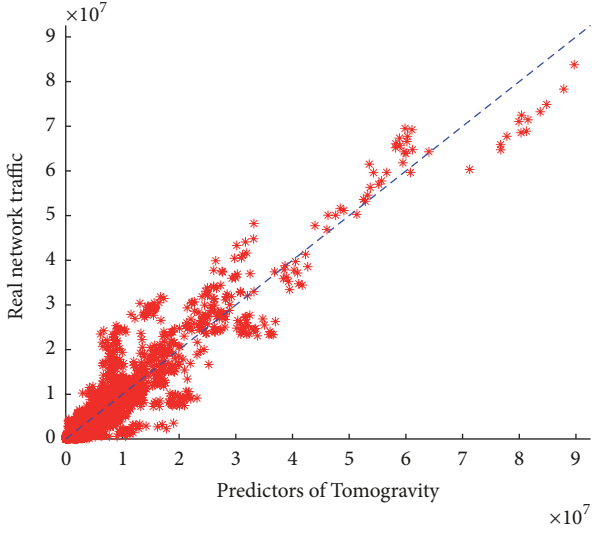
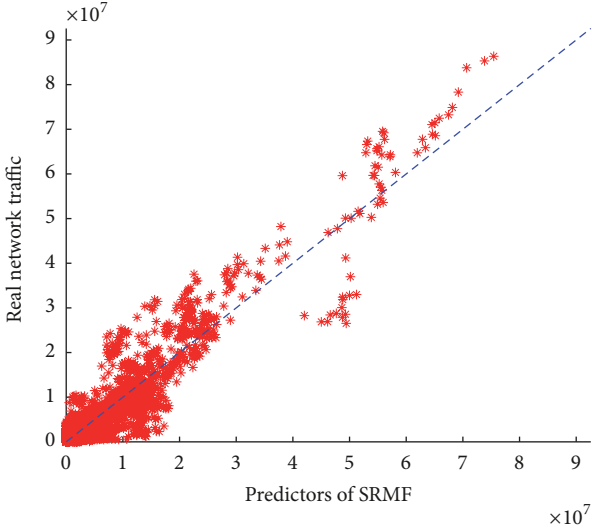FIGURE 9: Real traffic data versus their predictors via Tomogravity.

FIGURE 10: Real traffic data versus their predictors via SRMF.

our method has a consistently low SRE comparing with the other three methods. Figure 11(b) shows the TREs of four methods. From Figure 11(b), it shows that the TREs of PCA are much higher than that of the other methods. The TRE of DBNSTCS is the lowest one in four methods. The cumulative distributions of SRE and TRE are shown by Figure 12. It can show the prediction error more directly. Besides, we find that DBNSTCS has much more prominent improvement in TRE, comparing with SRE. That is because our method predicts the low-pass component of each traffic flow independently. In addition, contrasting to the high-pass component, this component has a significant effect on prediction. Hence, the weak improvement of SRE is caused by predicting the high-pass component using the Spatiotemporal Compressive Sensing method.

The diminutive error or bias of a prediction method does not mean it is available. Though it has low error, it fails to

provide precise predictors when it has high variance. Hence, the standard deviation is involved in our simulation as a metric for variance, which is defined as

$$\text{SD}(n) = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}\left(\widehat{x}_n(t) - x_n(t) - b(n)\right)^2}, \quad (12)$$

where $b(n) = (1/T)\sum_{t=1}^{T}(\widehat{x}_n(t) - x_n(t))$. In (12), T is the length of predicted traffic data set. Figure 13 shows the bias versus standard deviation of four methods. We find that the four methods perform very differently with respect to variance. Tomogravity has larger variance compared with the other methods. In contrast, the PCA method exhibits relatively high variance. The DBNSTCS and SRMF methods show relatively low variance.

Finally, the performance improvement ratio is shown in Figure 14 as an overall evaluation. The performance improvement ratio is defined as

$$\text{PIR} = \frac{\sum_{n=1}^{N^2}\sum_{t=1}^{T}\left|x_{n,a}(t)\right| - \sum_{n=1}^{N^2}\sum_{t=1}^{T}\left|x_{n,b}(t)\right|}{\sum_{n=1}^{N^2}\sum_{t=1}^{T}\left|x_{n,a}(t)\right|}, \quad (13)$$

where $x_{n,a}(t)$ and $x_{n,b}(t)$ denote the predictors via the algorithms a and b, respectively. The performance improvement ratios of DBNSTCS are 68.74%, 5.24%, and 14.70% to PCA, Tomogravity, and SRMF.

6. Conclusions and Future Work

This paper focuses on the problem of network traffic prediction in wireless mesh backbone networks and proposes a hierarchical prediction method. The proposed hierarchical prediction method divides the network traffic into two components and then predicts each component by different models. In detail, the proposed method takes advantage of DBN and Spatiotemporal Compressive Sensing for network traffic prediction. In our method, the DWT is applied to dividing the network traffic into two components, that is, the long-range dependence component and the fluctuation component represented by the low-pass and high-pass components, respectively. A deep architecture consisting of a DBN layer and a logistic regression architecture is proposed to predict the low-pass component. Meanwhile, the other is predicted by the SRMF method which can capture the spatiotemporal characteristic of the high-pass component. We assess the performance of the proposed prediction method and compare it with three methods that are widely used for network traffic prediction. According to the simulation, our method is well in prediction error, especially in TRE. The main bottleneck of wireless mesh backbone network traffic prediction is the predicted accuracy for the irregular fluctuations of network traffic. Thereby, the prediction algorithm aiming at low-pass components is necessary in the future.

Conflicts of Interest

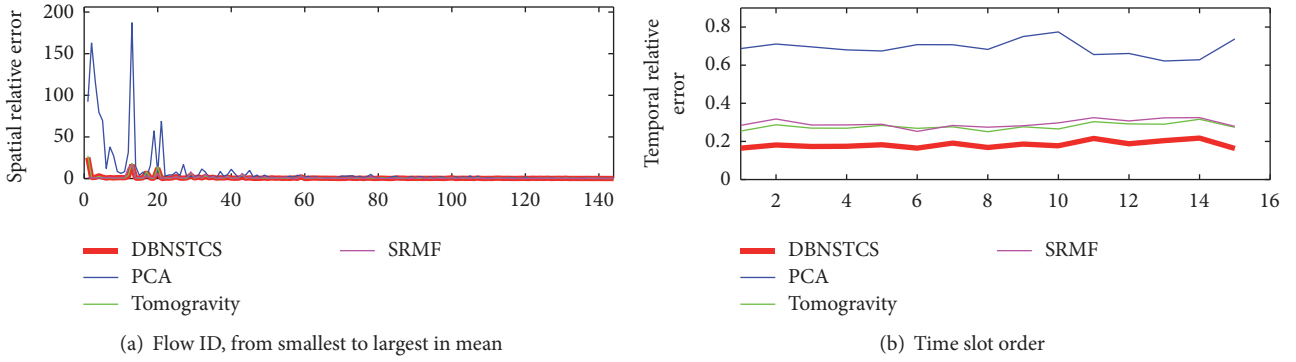The authors declare that they have no conflicts of interest.

(a) Flow ID, from smallest to largest in mean

(b) Time slot order

FIGURE 11: SRE and TRE.

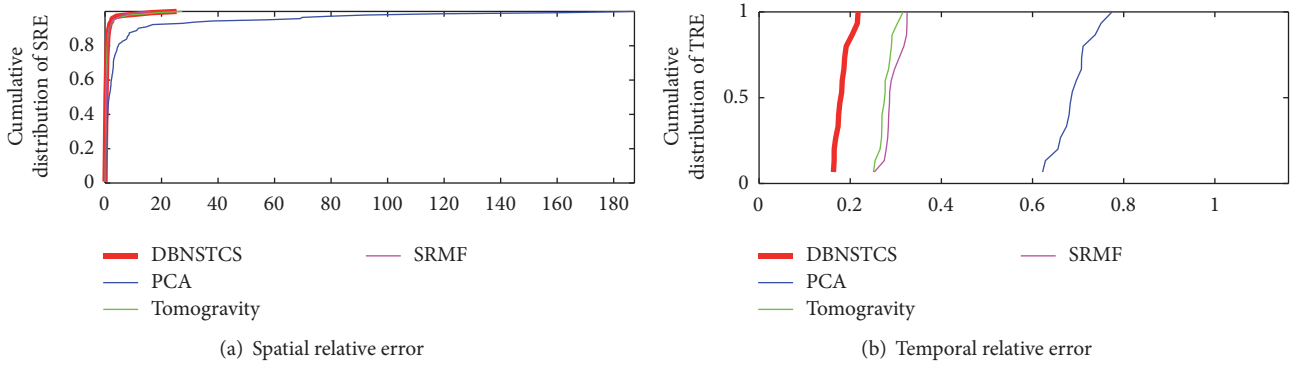(a) Spatial relative error

(b) Temporal relative error

FIGURE 12: Cumulative distributions of SRE and TRE.
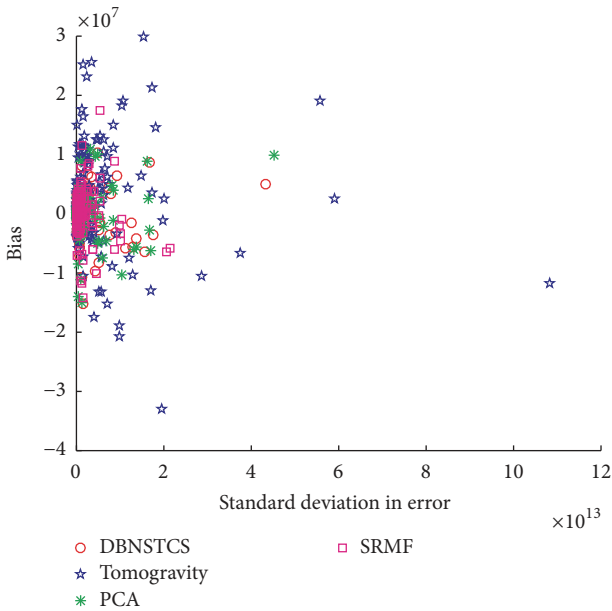
FIGURE 13: Bias versus standard deviation.
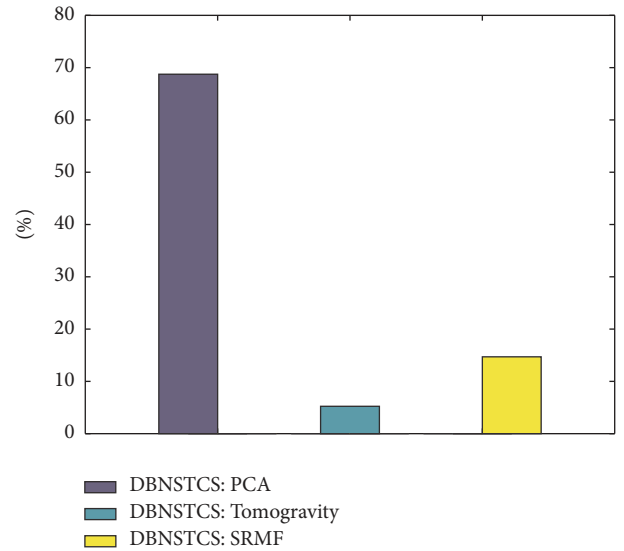
FIGURE 14: Performance improvement ratio.

Acknowledgments

References

[1] T. Qiu, R. Qiao, and D. Wu, "EABS: an event-aware backpressure scheduling scheme for emergency internet of things," *IEEE Transactions on Mobile Computing*, 2017.

[2] T. B. Reddy, B. S. Manoj, and R. Rao, "On the accuracy of sampling schemes for wireless network characterization," in

Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC 2008, pp. 3314–3319, usa, April 2008.

[3] T. Qiu, A. Zhao, F. Xia, W. Si, and D. O. Wu, "ROSE: Robustness Strategy for Scale-Free Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2944–2959, 2017.

[4] B. Zhou, D. He, and Z. Sun, "Traffic predictability based on ARIMA/GARCH model," in *Proceedings of the 2006 2nd Conference on Next Generation Internet Design and Engineering, NGI 2006*, pp. 200–207, Spain, April 2006.

[5] P. Kuanhoong, I. Tan, and C. YikKeong, "BitTorrent network traffic forecasting with ARIMA," vol. 4, pp. 143–156, 2012.

[6] T. Qiu, Y. Zhang, D. Qiao, X. Zhang, M. L. Wymore, and A. K. Sangaiah, "A Robust Time Synchronization Scheme for Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, pp. 1-1.

[7] P. Calyam and A. Devulapalli, "Modeling of multi-resolution active network measurement time-series," in *Proceedings of the 33rd IEEE Conference on Local Computer Networks, LCN 2008*, pp. 900–907, Canada, October 2008.

[8] J. Huang, C. Xing, and C. Wang, "Simultaneous wireless information and power transfer: technologies, applications, and research challenges," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 26–32, 2017.

[9] L. Nie, D. Jiang, S. Yu, and H. Song, "Network traffic prediction based on deep belief network in wireless mesh backbone networks," in *Proceedings of the 2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, San Francisco, CA, USA, March 2017.

[10] N. C. Anand, C. Scoglio, and B. Natarajan, "GARCH - Nonlinear time series model for traffic modeling and prediction," in *Proceedings of the NOMS 2008 - IEEE/IFIP Network Operations and Management Symposium: Pervasive Management for Ubiquitous Networks and Services*, pp. 694–697, Brazil, April 2008.

[11] J. Huang, Q. Duan, C.-C. Xing, and H. Wang, "Topology control for building a large-scale and energy-efficient internet of things," *IEEE Wireless Communications Magazine*, vol. 24, no. 1, pp. 67–73, 2017.

[12] M. Joshi and T. Hadi, *A review of network traffic analysis and prediction techniques*, Computer Science, 2015.

[13] Y. Liu, B. Li, X. Sun, and Z. Zhou, "A fusion model of SWT, QGA and BP neural network for wireless network traffic prediction," in *Proceedings of the 15th IEEE International Conference on Communication Technology (ICCT '13)*, pp. 769–774, IEEE, Guilin, China, November 2013.

[14] F. Xu, Y. Lin, J. Huang et al., "Big data driven mobile traffic understanding and forecasting: a time series approach," *IEEE Transactions on Services Computing*, vol. 9, no. 5, pp. 796–805, 2016.

[15] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.

[16] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[17] J. Huang, Q. Duan, Y. Zhao, Z. Zheng, and W. Wang, "Multicast Routing for Multimedia Communications in the Internet of Things," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 215–224, 2017.

[18] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," Tech. Rep., Technical University of Denmark, 2012.

[19] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices (Extended Version)," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 662–676, 2012.

[20] A. Soule, A. Lakhina, N. Taft et al., "Traffic matrices: Balancing measurements, inference and modeling," in *Proceedings of the SIGMETRICS 2005: International Conference on Measurement and Modeling of Computer Systems*, pp. 362–373, can, June 2005.

[21] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Computation of IP traffic from link," in *Proceedings of the ACM SIGMETRICS 2003 - International Conference on Measurement and Modeling of Computer Systems*, pp. 206–217, USA, June 2003.