Hindawi
Scientific Programming
Volume 2020, Article ID 2548351, 13 pages
https://doi.org/10.1155/2020/2548351

Research Article

A Peak Prediction Method for Subflow in Hybrid Data Flow

Zhaohui Zhang,[1,2,3] **Qiuwen Liu**,[1] **Ligong Chen**,[1] **and Pengwei Wang**[1]

[1]*School of Computer Science and Technology, Donghua University, Shanghai, China*
[2]*The Key Laboratory of Embeded System and Service Computing, Ministry of Education, Tongji University, Shanghai, China*
[3]*Shanghai Engineering Research Center of Network Information Services, Shanghai, China*

Correspondence should be addressed to Zhaohui Zhang; zhzhang@dhu.edu.cn

Received 24 October 2019; Accepted 10 January 2020; Published 14 February 2020

Guest Editor: Aibo Song

Copyright © 2020 Zhaohui Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Subflow prediction is required in resource active elastic scaling, but the existing single flow prediction methods cannot accurately predict the peak variation of subflow in hybrid data flow. These do not consider the correlation between subflows. The difficulty is that it is hard to calculate the correlation between different data flows in hybrid data flow. In order to solve this problem, this paper proposes a new method DCCSPP (subflow peak prediction of hybrid data flow based on delay correlation coefficients) to predict the peak value of hybrid data flow. Firstly, we establish a delay correlation coefficient model based on the sliding time window to determine the delay time and delay correlation coefficient. Next, based on the model, a hybrid data flow subflow peak prediction model and algorithm are established to achieve accurate peak prediction of subflow. Experiments show that our prediction model has achieved better results. Compared with LSTM, our method has decreased the MAE about 18.36% and RMSE 13.50%. Compared with linear regression, MAE and RMSE are decreased by 27.12% and 25.58%, respectively.

1. Introduction

The hybrid data flows are widely used in practical applications. For example, Alibaba's e-commerce platform uses a large-scale hybrid technology. This technology mixes online services with offline tasks. Hybrid data flow consists of online services and offline tasks. They enter the cluster at the same time and save the cost without affecting service quality.

The flow peak prediction is important in the active elastic expansion of the system [1]. Lombardi et al. [2] propose a novel elastic scaling approach, named ELYSIUM which contains the "predictionInputLoad" method to predict the maximum load. Bauer et al. [3] describe a new hybrid autoscaling mechanism, called Chameleon. Chameleon employs on-demand, automated time series-based forecasting methods to predict the arriving load intensity in combination. Hirashima et al. [4] give a new autoscaling mechanism which changes the scale of the target system based on the predicted workload.

In the active elastic scaling of the flow processing system, there are some studies on peak flow prediction. The authors regard network flow as a whole in the existing prediction methods. There are some traditional methods for network flow prediction, such as the ARIMA linear model and wireless network flow prediction model based on combinatorial optimization theory. Meanwhile, with the development in the neural network, the support vector machine (SVM) and other prediction model based on machine learning algorithm appears. Some authors use neural network models such as RNN [5], NARX recursive neural network model, LSTM [6], and GRU for predicting network peak flow. These prediction models can well explain the randomness and periodicity of flow.

However, the above methods are based on single flow prediction, without considering the possible correlation between individual flows in a hybrid data flow. Therefore, aiming at considering the influence of data correlation on peak flow prediction, this paper proposes a flow prediction method, named DCCSPP (subflow peak prediction of hybrid data flow based on delay correlation coefficients). We establish a delay correlation coefficient model to solve the correlation uncertainty of different subflows and consider

the correlation influence between subflows based on the predicting results of the single flow. The more accurate the prediction of flow peaks, the more reliable the system flow information will be obtained, and this will provide better indexing parameters for the system's elastic scaling.

2. Related Work

In recent years, flow predictions based on time series have always been an attractive research area. Developing predictive models plays an important role in interpreting complex real-world elements [7].

Many of the traditional learning methods are used for time series prediction. Zhang et al. [1] propose an agile perception method to predict abnormal behavior. Yu et al. [8] describe an ARIMA linear model to predict network flow sequence. Aiming at solving the problem that a single model cannot fully describe change characteristics, a wireless network flow prediction model based on combinatorial optimization theory is proposed by Chen and Liu [9]. Liu et al. [10] give online learning algorithms for estimating ARIMA models under relaxed assumptions on the noise terms. Adebiyi et al. [11] examine the forecasting performance of ARIMA and artificial neural networks model. Wu and Wang [12] investigate time series prediction algorithms by using a combination of nonlinear filtering approaches and the feedforward neural network (FNN). Joo and Kim [13] propose a forecasting method based on wavelet filtering. Han et al. [14] introduce a multioutput least square support vector regressor. Chandra and Al-Deek [15] discuss a vector autoregressive model for prediction at short-term flow prediction on freeways. Conventional techniques for time series prediction are limited in their ability to process big data with high dimensionality, as well as efficiently represent complex functions. If the amount of linear data are not too large, the statistical method is reliable enough to be used for prediction. At the same time, the generated model is very complex and difficult to be implemented by nonlinear data types, so the prediction results are not very accurate when there are massive data.

Deep learning-based models have been successfully applied in many fields to time series prediction. There are many prediction models, which based on machine learning have been proposed. Haviluddin and Alfred [16] introduce a NARX recursive neural network model to predict network flow. Nie et al. [17] propose a novel network flow prediction method based on deep belief network (DBN) and logistic regression model for network flow prediction. In [18], network flow prediction of neural network models such as RNN [5], LSTM [6], and GRU is used. Hoermann et al. [19] report a deep CNN model for dynamic occupancy grid prediction with data from multiple sensors. The advantage of a Gaussian processes lies in its ability of modeling the uncertainty hidden in data, which is provided by predicting distributions [20]. Deep learning-based models are good at discovering intricate structure in large data sets [7]. These prediction models can well explain the randomness and periodicity of flow.

As mentioned above, the above methods are all for single flow prediction, without considering the possible correlation between data flows in hybrid flow. However, in the hybrid data flow, there is a lack of research on such flow prediction. Therefore, this paper mainly studies the correlation between different subflows in the hybrid flow and the peak prediction of each subflow.

3. Delay Correlation Coefficient Model Based on Sliding Time Window

In hybrid data flows, there are different degrees of correlation between different subflows. Considering the correlation between subflows and the pseudocorrelation caused by time analysis, this paper proposes a delay correlation coefficient model, which adds sliding time window according to Pearson correlation coefficient and time difference analysis [21]. This model is to calculate the delay correlation coefficient and delay time difference between different subflows. Based on the delay coefficient, the data flow that has an influence on the target subflow prediction is filtered out.

Correlation analysis [21] refers to the measure the closeness of the variables between two or more related variable elements. Correlation elements need to have a certain connection or probability to conduct correlation analysis.

The Pearson correlation coefficient, also known as Pearson product-moment correlation coefficient, represents the linear correlation between the two sets of variables X and Y. The formula is shown as follows:

$$\operatorname{cov}(X,Y) = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{n-1}. \tag{1}$$

Equation (1) is the covariance formula. The covariance is divided by the standard deviation of the two related variables to obtain the Pearson correlation coefficient, which is described in formula (2). It is to compensate for the weak representation of the covariance value in the degree of random variable correlation:

$$\rho(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}. \tag{2}$$

The Pearson correlation coefficient can always be between $[-1,1]$. The closer the coefficients are to the extremes at both ends, the greater the linear relationship between the two random variables. If the coefficient is close to 0, it means that the two variables are not linearly related. If the coefficient approaches 1, it means that X and Y can be well described by the straight line equation, all data points fall well on a straight line, and X increases as Y increases. The coefficient approaching -1 means that all data points fall on a straight line, and X decreases as Y increases.

In the flow processing system, the input of data is generally composed of multiple subflows, which we call it a hybrid data flow. This article defines the hybrid data flow as follows

Definition 1. The hybrid data flow in the k period is $S_k = [(t_i, a_j) \mid 1 \le j \le n, 1 \le i \le k, n \le k]$, where n indicates that there are n kinds of data flows and (t_i, a_j) indicates that data belonged to the jth data flow arrives system at the time of t_i.

Definition 2. The data set constituting a business is $U = \{a_1, a_2, \ldots, a_z\}$, where z indicates that the data set of the service consists of z kinds of data flows. Thus, service correlation exists in these data. For example, a hybrid data flow consisting of device login information and user behavior information. The flow of user behavior information is affected by the flow of device login information, and the two have a partial-order relationship. Since different service data flows require different processing operations and computing resources, it is necessary to perform shunt operations on the data of the hybrid data flow, as shown in Figure 1.

Through the statistics of discrete hybrid data, the observation sequence of each subflow is obtained. A set of hybrid data flow observation sequences composed of subflow observation sequences are defined.

Definition 3. The hybrid data flow observation sequence set is $M = \{m_1, m_2, \ldots, m_n\}$, where n represents M contains n data flows. m_i represents the observed sequence of the ith data flow in M, that is, $m_i = \{m_i^1, m_i^2, \ldots, m_i^t, \ldots, m_i^l\}$, where m_i^t represents the observed value of data flow m_i at t time and l represents the l observation values of the data flow m_i. m_j represents the observation sequence of the jth data flow in M, that is, $m_j = \{m_j^1, m_j^2, \ldots, m_j^t, \ldots, m_j^l\}$, where m_j^t represents the observed value of data flow m_j at time t and l represents l observations in the data flow m_j. And $i \neq j$.

Definition 4. The ith subflow in hybrid data flow m_i.

Definition 5. The delay time e is shown in Figure 2. It means that the change of m_j in time $t - e$ has an effect on m_i at time t.

Definition 6. The size of the sliding time window is h, as shown in Figure 3.

Let $X = m_i$, where $X = \{x_1, x_2, \ldots, x_t, \ldots, x_l\}$, so $x_t = m_i^t$. Let $Y = m_j$, where $Y = \{y_1, y_2, \ldots, y_t, \ldots, y_l\}$, so $y_t = m_j^t$.

Definition 7. The correlation coefficient of m_i and m_j when the delay time is $d\rho(m_i, m_j)_e$. The calculation formula of $d\rho(m_i, m_j)_e$ is described in the following formula:

$$d\rho\left(m_i, m_j\right)_e = \frac{1}{l-h-e}\sum_{t=0}^{l-h-e}\left|\rho\left(X_{t-1}, Y_{t-e-1}\right)\right|, \tag{3}$$

where $X_{t-1} = \{x_{t-h}, x_{t-h+1}, \ldots, x_{t-1}\}$ and $X_{t-1} \subseteq X$. $Y_{t-e-1} = \{y_{t-h-e}, y_{t-h-e+1}, \ldots, y_{t-e-1}\}$ and $Y_{t-e-1} \subseteq Y$. X_{t-1} and Y_{t-e-1} are shown in Figure 4.

Definition 8. The maximum delay correlation coefficient between m_i and m_j is $\max[d\rho(m_i, m_j)]$. Its calculation formula is as follows:

$$\max\left[d\rho\left(m_i, m_j\right)\right] = \max\left[d\rho\left(m_i, m_j\right)\right]_e, \quad e \in [1, l-2h] \text{ and } e \in N. \tag{4}$$

When predicting m_i, it is necessary to select the data flow m_k ($1 \leq k \leq n$ and $k \neq i$) with the highest delay correlation for the auxiliary prediction. The selection formula of m_k is as follows:

$$\max\left[d\rho\left(m_i, m_k\right)\right] = \max\left\{\max\left[d\rho\left(m_i, m_j\right)\right]\right\}, \quad j \in [1, n],\ j \in N,\ j \neq i. \tag{5}$$

Algorithm 1 gives the pseudocode for selecting the auxiliary data flow algorithm as follows.

4. Hybrid Data Flow Subflow Peaking Prediction Model

The selected data flow m_i (i.e., X) is separately predicted by a single flow prediction method, and an initial prediction result set $X' = \{x_1', x_2', \ldots, x_t', \ldots, x_l'\}$ of X is obtained, where x_t' represents an initial prediction result for the value x_t at time t in X.

Definition 9. The variation in x at time t is Δx_t. Δx_t represents the difference between the single prediction result at time t and time $t - 1$. The calculation formula is as follows:

$$\Delta x_t = x_t' - x_{t-1}. \tag{6}$$

Definition 10. The amount of change in y at time t is Δy_t. Δy_t represents the difference between the observed value at time $t - e$ and $t - e - 1$. The calculation formula is as follows:

$$\Delta y_t = y_{t-e} - y_{t-e-1}. \tag{7}$$

Definition 11. To scale the range of y to the range of x in a same level, we defined pro_{t-1}, which is described as follows:

$$\text{pro}_{t-1} = \frac{\max\left(X_{t-1}\right) - \min\left(X_{t-1}\right)}{\max\left(Y_{t-1-e}\right) - \min\left(Y_{t-1-e}\right)}. \tag{8}$$

Definition 12. At the time t, the final prediction result of x_t is x_t''. The calculation formula is as follows:

$$x_t'' = \Delta x_t * \alpha + \frac{\rho\left(X_{t-1}, Y_{t-1-e}\right)}{\left|\rho\left(X_{t-1}, Y_{t-1-e}\right)\right|} * \Delta y_t * (1-\alpha) * \text{pro}_{t-1}, \tag{9}$$

where α represents the weight of the correlation coefficient, and the calculation formula is as follows:

$$\alpha = \frac{1}{1 + \left|\rho\left(X_{t-1}, Y_{t-1-e}\right)\right|}. \tag{10}$$

Algorithm 2 gives the pseudocode for the hybrid data flow correlation prediction algorithm as follows.

The evaluation indexes in this paper are root mean square error (RMSE) and mean absolute error (MAE). The calculation formulas are as follows:

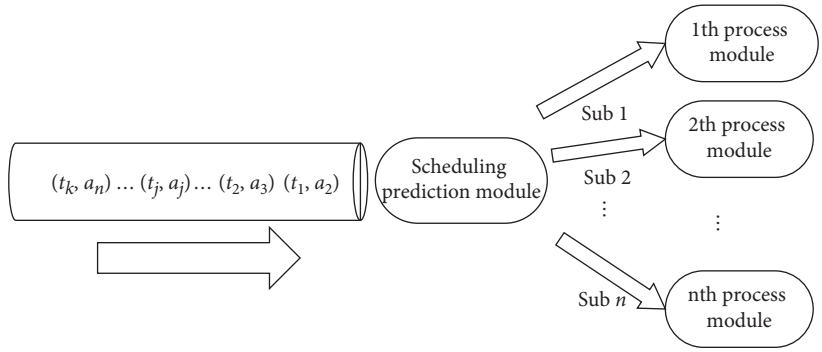

Figure 1: Split flow diagram.

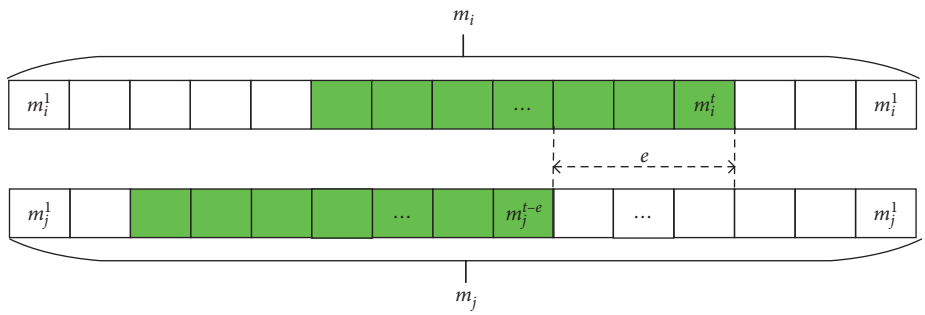

Figure 2: Delay time.

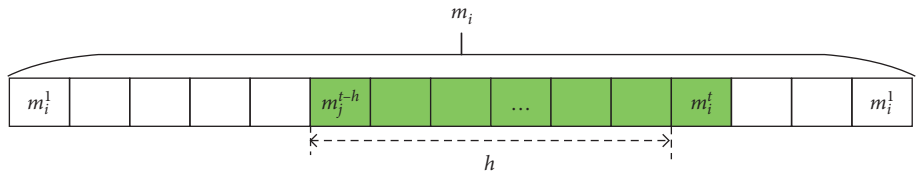

Figure 3: Sliding time window h.

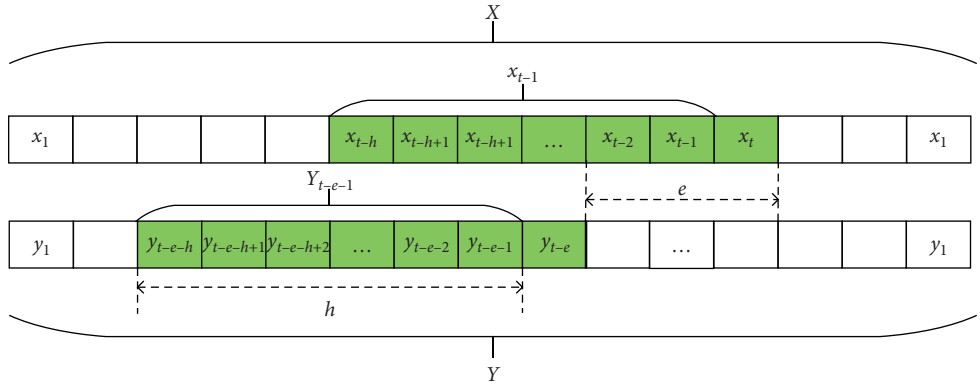

Figure 4: X_{t-1} and Y_{t-e-1} schematic.

Input: the list of steams; the size of window; the number of predicted steam
Output: the number of subsidiary steam; the number of delay time
(1) **procedure** chooseSteam ()
(2) **for** iterate through the list of steams **do**
(3) **for** iterate through all of the number of delay time **do**
(4) **for** iterate through all of the size of window **do**
(5) Calculate and get the correlation coefficient between the delay time and window
(6) Summing up the correlation coefficients
(7) Calculate and get the mean of the correlation coefficient
(8) Update the maximum delay correlation coefficient and the delay time
(9) Update the maximum delay correlation coefficient and the delay time
(10) Update the number of the auxiliary data flow
(11) **return** the number of the auxiliary data flow and the delay time

Algorithm 1: Choose flow.

Input: the list of predicted steam; the list of first prediction flow; the list of subsidiary steam; the number of delay time; the size of window; the number of time
Output: the number of the final predicted value at time t
(1) **procedure** prediction ()
(4) Calculate and get Δx_t based on formula (6)
(5) Calculate and get Δy_t based on formula (7)
(6) Calculate and get pro_{t-1} based on formula (8)
(7) Calculate and get α based on formula (10)
(8) Calculate and get the final prediction result based on formula (9)
(9) **return** the final prediction result

Algorithm 2: Confounding flow correlation prediction.

$$\text{RMSE} = \sqrt{\frac{1}{l}\sum_{t=1}^{l}\left(x_t'' - x_t\right)^2},$$
$$\text{MAE} = \frac{1}{l}\sum_{t=1}^{l}\left|x_t'' - x_t\right|. \tag{11}$$

The smaller the mean absolute error index value is, the more accurate the prediction result is. The smaller the root mean square error value is, the fewer the abnormal discrete points are, and the higher the prediction accuracy is.

5. Experimental Verification

5.1. Data Set. In order to analyze the prediction performance of the prediction method proposed in this paper, the device login data and behavior acquisition data provided by the mobile phone APP of a credit company in three periods of three months are selected. We collect 13,567 pieces of equipment login data and 282,685 pieces of behavioral data in a certain period of June, as data set 1, as shown in Figures 5 and 6. There are 27,381 device login records and 344,109 behavior data selected in a certain period of July, as data set 2, as shown in Figures 7 and 8. And data set 3 selects 17550 device login records and 755693 behavior data in a certain period of November, as shown in Figures 9 and 10.

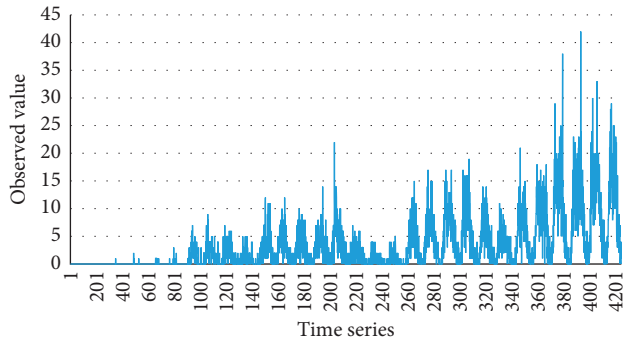

Figure 5: Data set 1 device login statistics.

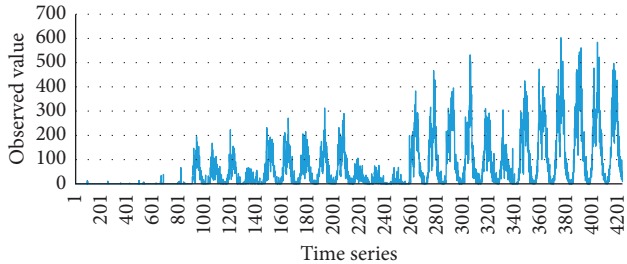

Figure 6: Data set 1 behavior collection statistics.

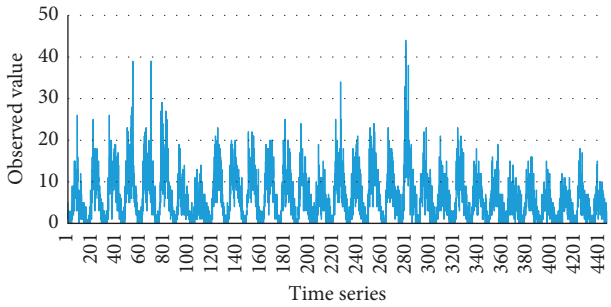

Figure 7: Data set 2 device login statistics.

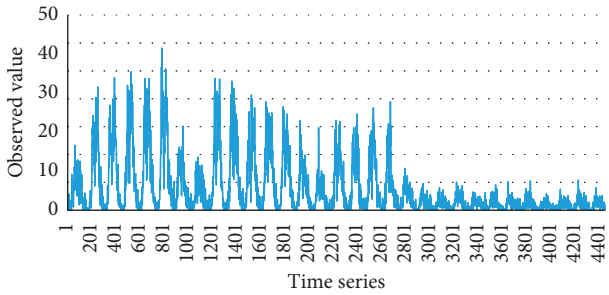

Figure 8: Data set 2 behavior collection statistics.

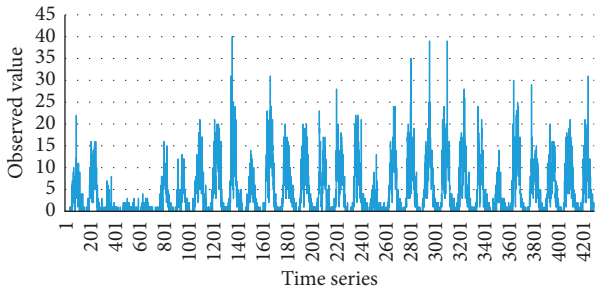

Figure 9: Data set 3 device login statistics.

Each subset selects 4465 observations. From Figures 5–10, we can see that the change trend of device login statistics and behavior collection statistics is close, and there is a correlation between them. In the experiment, firstly, the results predicted by LSTM and unary linear regression model are as the control group. Then, the results by our model are as the experimental group. In the end, compare their prediction indicators and error indicators of peak prediction.

5.2. Compared with LSTM Prediction Method. In this paper, the first 90% observed values of each data set is selected as training sets to train the LSTM learning model, and the last 10% is used as the test set to analyze the predictive ability of the model. The overall prediction results of the test sets of data set 1, data set 2, and data set 3 are obtained, as shown in Figures 11–13. And the prediction results for a period with high observed values in data set 1, data set 2, and data set 3 are shown in Figures 14–16. In the DCCSPP, it is necessary

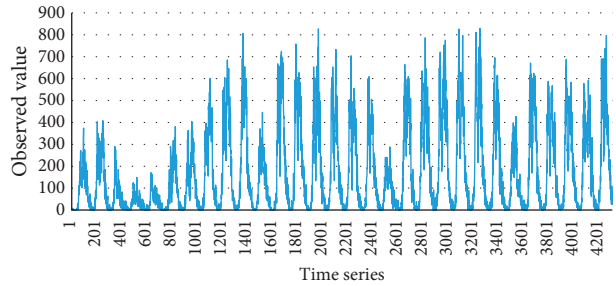

Figure 10: Data set 3 is behavior collection statistics.

to intercept the observation value of time window size for calculation, so before 90, the prediction method cannot give the prediction result, and the value is 0.

In this paper, we need to discuss the influence of time window, and the results of experiment on data set 2 are shown in Figure 17.

Compared with the LSTM model, it can be seen from Figures 14–16 that the results changes in DCCSPP are closer to the real-observed values.

It can be seen from Figure 17 that the selection of time window has certain influence on the prediction results. Too small or too large time window has a bad influence on the prediction results. Therefore, in addition to data set 3, this article selects 90 as the size of the time window. On data set 3, the prediction method can get better prediction results when the time window size is 240.

The errors of prediction results for data set 1, data set 2, and data set 3 in this paper are shown in Table 1. The prediction method has the most obvious improvement in data set 1. MAE and RMSE decreased by 13.46% and 17.80%, respectively. And we found that the smaller values of the test set of data set 2 lead that the MAE and RMSE of data set 2 are smaller than the others. In the end, the overall results show that the accuracy of the prediction results can be improved by using the correlation coefficient algorithm based on the prediction results of the LSTM model.

This paper compares the calculation indexes of prediction results of multiple maximum peak points in data set 1, data set 2, and data set 3, and the results are shown in Table 2. It illustrates that the peak prediction in the test set is not accurate due to the unfavorable data in the training set of data set 1. The method proposed in this paper can significantly improve the index of peak prediction, with MAE and RMSE increasing by 41.46% and 33.79%, respectively. In data set 2 and data set 3, MAE is decreased about 12.83% averagely. However, the improvement in the RMSE index was limited, with an average increase of 3.3%. In conclusion, the method proposed in this paper can improve the final peak prediction results.

5.3. Compared with Simple Linear Regression. In this paper, the unary linear regression model is used to predict the test sets in data set 1, data set 2, and data set 3. Through experiments, the prediction results of data set 1, data set 2, and data set 3 are shown in Figures 18–20, respectively. The

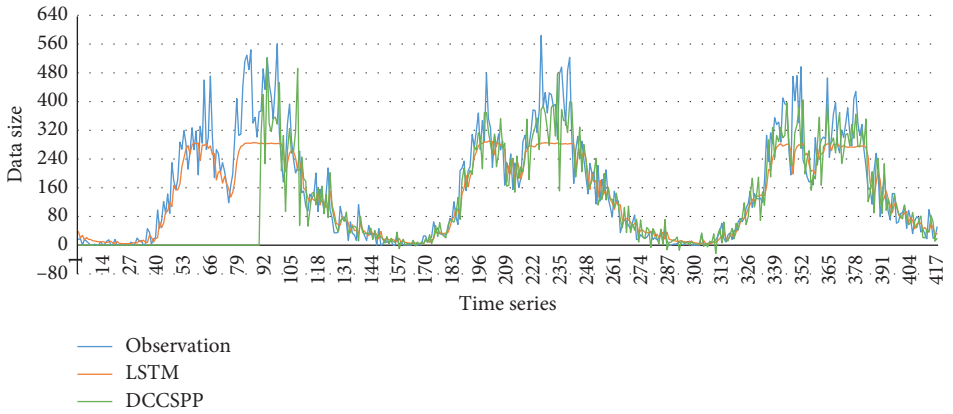

Figure 11: Comparison graph of predicted results on data set 1.

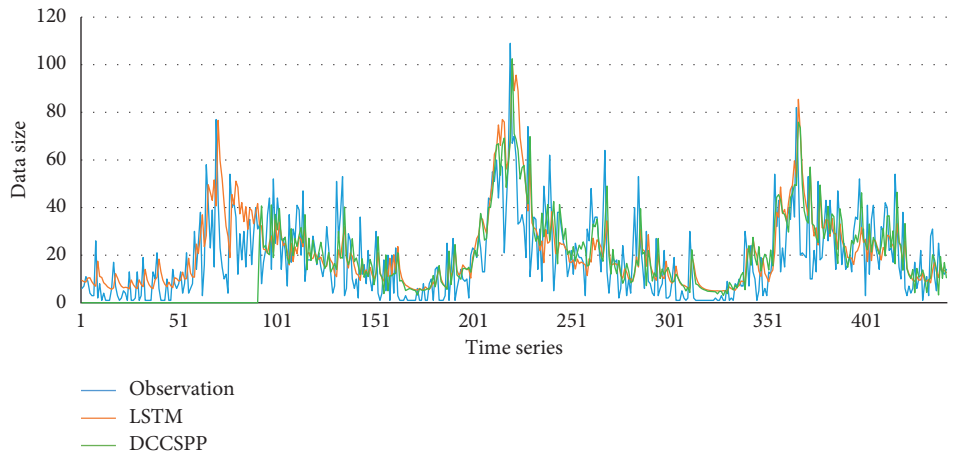

Figure 12: Comparison graph of predicted results on data set 2.

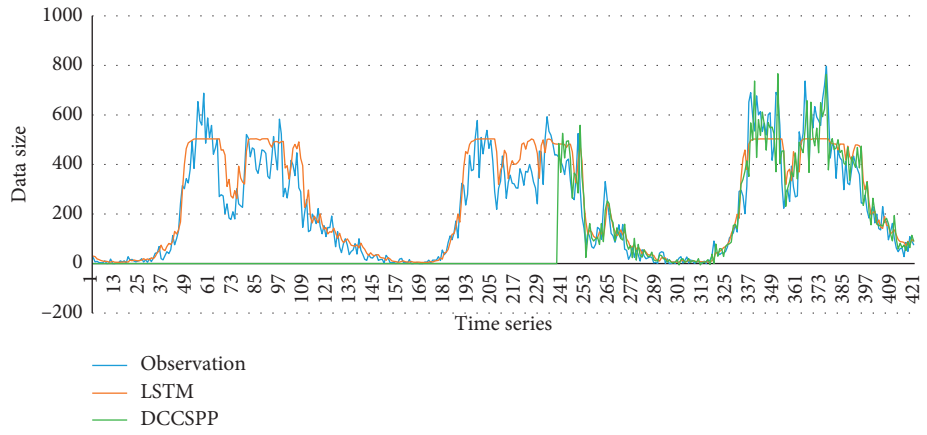

Figure 13: Comparison graph of predicted results on data set 3.

prediction results for a period with high observation values are shown in Figures 21–23, respectively.

It can be concluded from Figures 21–23 that the results of the prediction model in this paper are closer to the actual changes in the observations compared to the unitary linear regression model.

In this paper, the error comparison of prediction results for data set 1, data set 2, and data set 3 is shown in Table 3. It

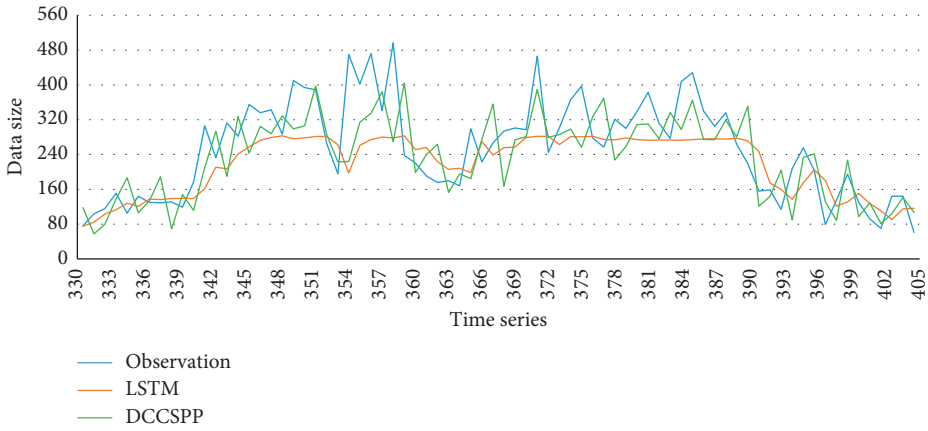

Figure 14: Comparison graph of prediction results for a time period in data set 1.

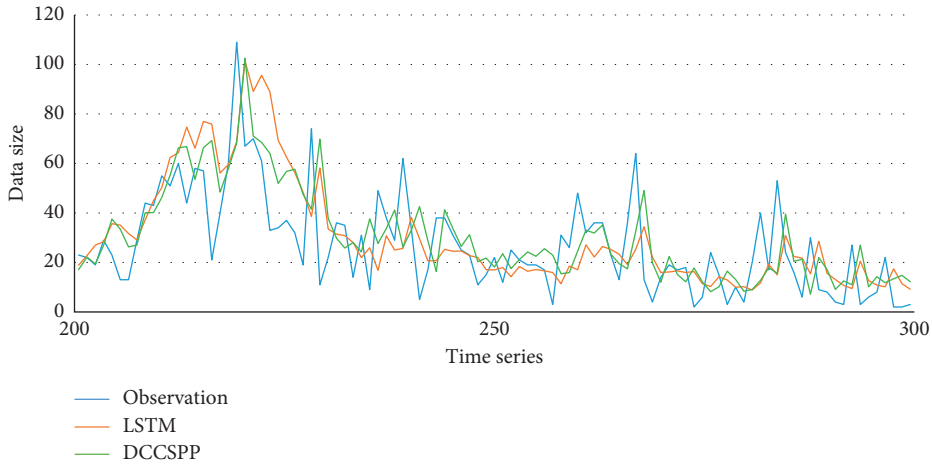

Figure 15: Comparison graph of prediction results for a time period in data set 2.

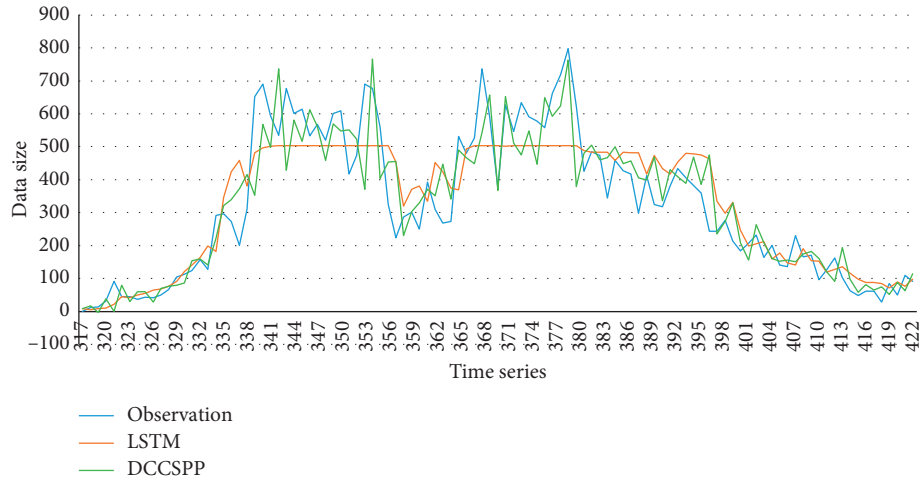

Figure 16: Comparison graph of prediction results for a time period in data set 3.

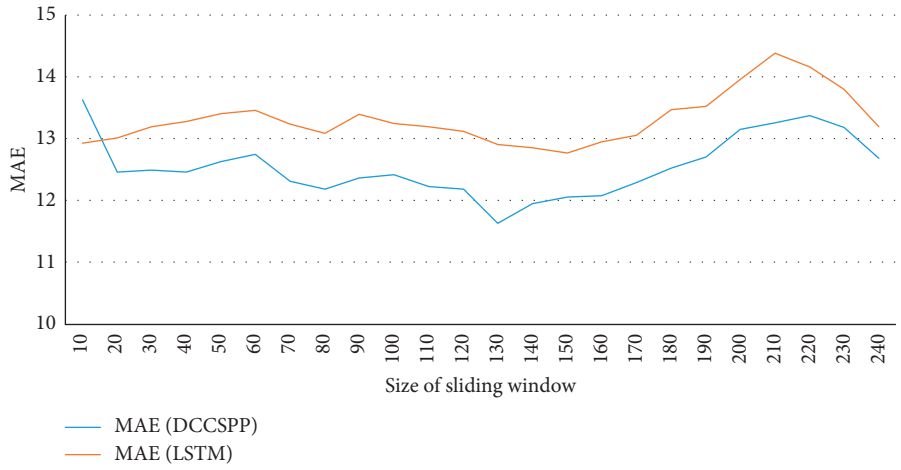

Figure 17: Comparison graph of average absolute errors of different time windows.

Table 1: Error comparison of prediction results.

Data set	Rate	LSTM	New	Improved (%)
Data set 1	MAE	68.85	59.58	13.46
	RMSE	98.32	80.83	17.80
Data set 2	MAE	13.39	12.36	7.73
	RMSE	17.34	16.13	7.00
Data set 3	MAE	59.57	53.86	9.59
	RMSE	90.98	87.01	4.36

Table 2: Peak prediction index decreased.

Data set	Number of peak points	Improved MAE (%)	Improved RMSE (%)
Data set 1	13	41.46	33.79
Data set 2	8	12.63	6.54
Data set 3	11	13.03	0.16

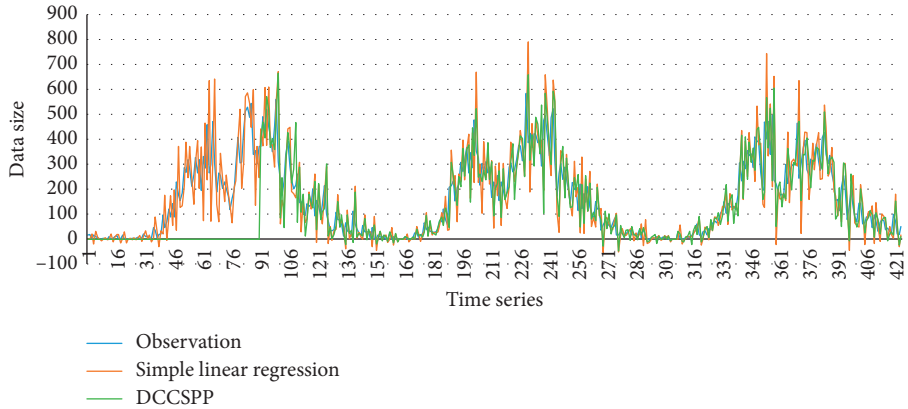

Figure 18: Comparison graph of predicted results on data set 1.

can be seen from the chart that the prediction results of the unary linear regression model are not as good as the LSTM model in MAE and RMSE indexes. Through the method proposed in this paper, the prediction result index on data set 1 is better than LSTM. The method proposed in this paper is used in the unary linear regression prediction model. The

Figure 19: Comparison graph of predicted results on data set 2.

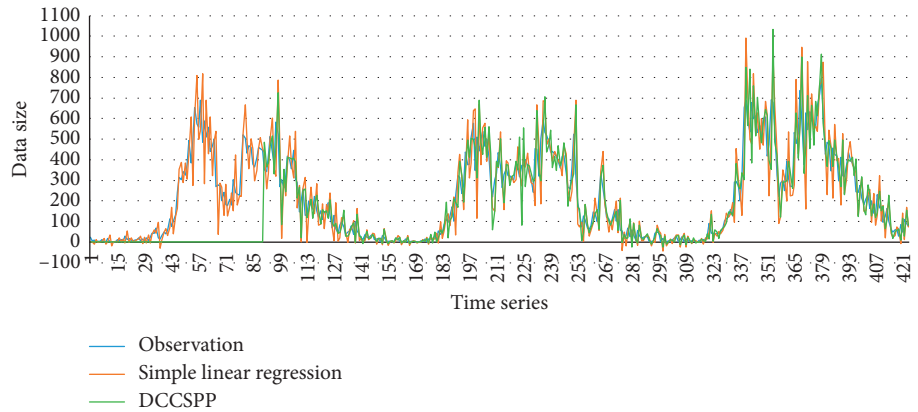

Figure 20: Comparison graph of predicted results on data set 3.

Figure 21: Comparison graph of prediction results for a time period in data set 1.

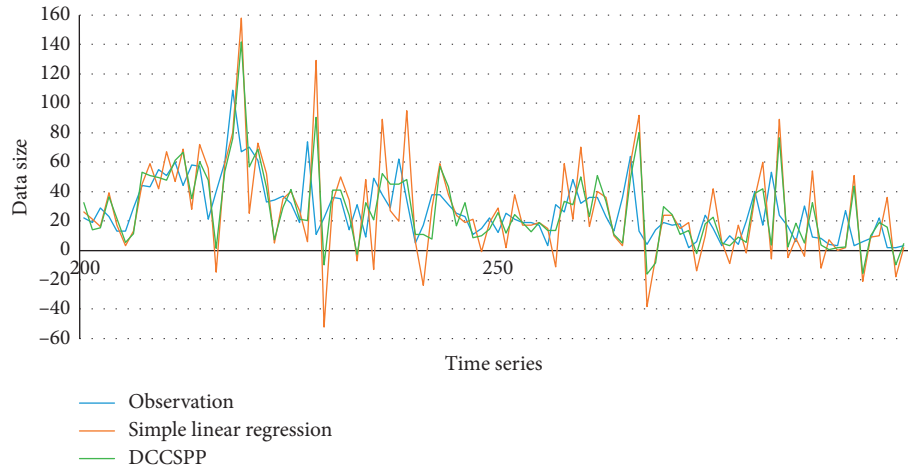

Figure 22: Comparison graph of prediction results for a time period in data set 2.

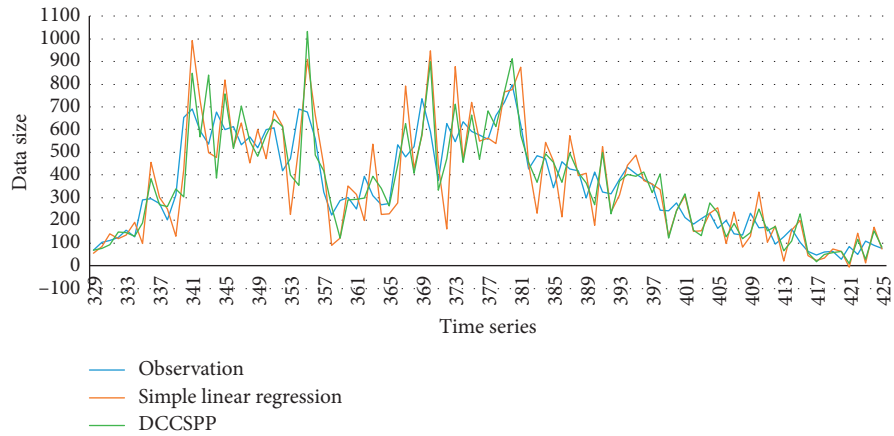

Figure 23: Comparison graph of prediction results for a time period in data set 3.

Table 3: Error comparison of prediction results.

Data set	Rate	Simple linear regression	New	Improved (%)
Data set 1	MAE	75.21	63.60	15.44
	RMSE	102.58	87.01	16.90
Data set 2	MAE	21.17	15.67	26.00
	RMSE	27.19	20.11	26.03
Data set 3	MAE	80.73	67.09	16.90
	RMSE	121.77	95.17	21.84

Table 4: Peak prediction index decreased.

Data set	Number of peak points	Improved MAE (%)	Improved RMSE (%)
Data set 1	13	33.45	28.73
Data set 2	8	32.40	29.49
Data set 3	11	15.50	18.52

experimental results show that the MAE value and the RMSE value are decreased by 15% to 26%. In conclusion, the method proposed in this paper used in the unary regression model can greatly improve the accuracy of the prediction results.

This paper compares the prediction results of multiple maximum peak points in data set 1, data set 2, and data set 3, and the results are shown in Table 4. As can be seen from the chart, 13 peak points with the highest observed values are selected in data set 1 to calculate the improvement of MAE and RMSE. They increase 33.45% and 28.73%, respectively. And 8 peak points with the highest observed values are selected in data set 2 to calculate the improvement of MAE and RMSE. They improve 32.40% and 29.49%, respectively. In data set 3, the 11 peak points with the highest observed values are selected to calculate the MAE and RMSE, which increase 15.50% and 18.52%, respectively. In conclusion, the method proposed in this paper can improve the final peak prediction results in the single-variable linear regression model's peak prediction results.

The chart information of experiment 1 and experiment 2 can be obtained. The method proposed in this paper can improve the prediction results in both overall prediction and peak prediction. Compared with the LSTM method, MAE

and RMSE decreased by 18.36% and 13.50%, respectively. Compared with the unary linear regression method, MAE and RMSE decreased by 27.12% and 25.58%, respectively. In the overall forecast, MAE and RMSE rose about 14.85% and 15.66%, respectively. In the peak forecast, MAE and RMSE decreased by about 24.75% and 19.54%, respectively. Therefore, the peak prediction method of hybrid data subflow proposed in this paper can effectively improve the result based on the prediction result.

6. Conclusions

For the hybrid data flow, there are related uncertainties in each subflow at different times. This paper establishes the delay correlation coefficient model. Through this model, the delay correlation coefficient and delay time are calculated. The prediction results of the respective flows are calculated by using the peak prediction method in the hybrid data flow. Experiments show that the DCCSPP model has good prediction results when there is uncertainty between the subflows in the hybrid flow.

In future work, we will introduce the correlation between subflows into the machine learning model. Using machine learning methods improves the accuracy of delay correlation coefficient calculations and the prediction results. At the same time, the model can also be applied to dynamic hybrid data flows. Design a dynamic allocation scheme based on the predicted peak results of each subflow, dynamically allocating resources to systems that require elastic scaling.

Data Availability

The data used in the paper came from an insurance company of China. Subject to the confidentiality agreement, the experimental data set cannot be disclosed to the public, and the name of the company cannot be mentioned in the paper. However, we guarantee that the data set used is authentic with the company.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of Shanghai (no. 19ZR1401900), Shanghai Science and Technology Innovation Action Plan Project (no. 19511101802), and National Natural Science Foundation of China (nos. 61472004 and 61602109).

References

[1] Z. Zhang and J. Cui, "An agile perception method for behavior abnormality in large-scale network service systems," *Chinese Journal of Computers*, vol. 40, no. 2, pp. 503–519, 2017, in Chinese.

[2] F. Lombardi, L. Aniello, S. Bonomi, and L. Querzoni, "Elastic symbiotic scaling of operators and resources in stream processing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 3, pp. 572–585, 2017.

[3] A. Bauer, N. Herbst, S. Spinner, A. Ali-Eldin, and S. Kounev, "Chameleon: a hybrid, proactive auto-scaling mechanism on a level-playing field," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 800–813, 2018.

[4] Y. Hirashima, K. Yamasaki, and M. Nagura, "Proactive-reactive auto-scaling mechanism for unpredictable load change," in *Proceedings of the 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pp. 861–866, IEEE, Kumamoto, Japan, July 2016.

[5] R. Madan and P. SarathiMangipudi, "Predicting computer network traffic: a time series forecasting approach using DWT, ARIMA and RNN," in *Proceedings of the Eleventh International Conference on Contemporary Computing (IC3)*, pp. 1–5, IEEE, Noida, India, August 2018.

[6] Y. Tian and L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," in *Proceedings of the IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, pp. 153–158, IEEE, Chengdu, China, December 2015.

[7] Z. Han, J. Zhao, H. Leung, K. F. Ma, and W. Wang, "A review of deep learning models for time series prediction," in *IEEE Sensors Journal*, pp. 1–1.

[8] Y. Yu, J. Wang, M. Song et al., "Network traffic prediction and result analysis based on seasonal ARIMA and correlation coefficient," in *Proceedings of the International Conference on Intelligent System Design and Engineering Application*, vol. 1, pp. 980–983, IEEE, Changsha, China, October 2010.

[9] H. Chen and J. Liu, "Modeling and forecast of wireless network traffic based on combinatorial optimization theory," *Modern Electronics Technique*, vol. 39, no. 23, pp. 43–47, 2016.

[10] C. Liu, S. C. H. Hoi, P. Zhao et al., "Online arima algorithms for time series prediction," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, AZ, USA, February 2016.

[11] A. A. Adebiyi, A. O. Adewumi, and C. K. Ayo, "Comparison of ARIMA and artificial neural networks models for stock price prediction[J]," *Journal of Applied Mathematics*, vol. 2014, Article ID 614342, 7 pages, 2014.

[12] X. Wu and Y. Wang, "Extended and Unscented Kalman filtering based feedforward neural networks for time series prediction," *Applied Mathematical Modelling*, vol. 36, no. 3, pp. 1123–1131, 2012.

[13] T. W. Joo and S. B. Kim, "Time series forecasting based on wavelet filtering," *Expert Systems with Applications*, vol. 42, no. 8, pp. 3868–3874, 2015.

[14] Z. Han, Y. Liu, J. Zhao, and W. Wang, "Real time prediction for converter gas tank levels based on multi-output least square support vector regressor," *Control Engineering Practice*, vol. 20, no. 12, pp. 1400–1409, 2012.

[15] S. R. Chandra and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models," *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 53–72, 2009.

[16] Haviluddin and R. Alfred, "Performance of modeling time series using nonlinear autoregressive with eXogenous input (NARX) in the network traffic forecasting," in *Proceedings of the International Conference on Science in Information Technology*, IEEE, Yogyakarta, Indonesia, October 2016.

[17] L. Nie, D. Jiang, L. Guo, S. Yu, and H. Song, "Traffic matrix prediction and estimation based on deep learning for data center networks," in *Proceedings of the Globecom Workshops*, IEEE, Washington, DC, USA, December 2017.

[18] R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Applying deep learning approaches for network traffic prediction," in *Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 2353–2358, IEEE, Udupi, India, September 2017.

[19] S. Hoermann, M. Bach, and K. Dietmayer, "Dynamic occupancy grid prediction for urban autonomous driving: a deep learning approach with fully automatic labeling," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2056–2063, IEEE, Brisbane, Australia, May 2018.

[20] C. K. I. Williams and C. E. Rasmussen, *Gaussian Processes for Machine learning*, MIT press, Cambridge, MA, USA, 2006.

[21] H. Wang, Z. Zhang, and P. Wang, "A situation analysis method for specific domain based on multi-source data fusion," in *Proceedings of the International Conference on Intelligent Computing*, pp. 160–171, Springer, Wuhan, China, August 2018.