International Journal of Database Theory and Application
Vol.9, No.11 (2016), pp.195-206
http://dx.doi.org/10.14257/ijdta.2016.9.11.18

Network Traffic Anomaly Detection Based on N-ARMA Model

Pingping Gu[1], Shijing Zhang[2], Zhimin Huang[2] and Qingfeng Wu[*3]

[1]*Computer Science Department, Tan Kah Kee College Xiamen University, China*
E-mail:gpp1011@xujc.com
[2]*Software School, Xiamen University, Xiamen, China*
[*3,] *Corresponding Author Software School, Xiamen University, China*
E-mail: qfwu@xmu.edu.cn

Abstract

With the rapid development of the Internet and the continuous expanding of the data network, little potential anomaly can seriously affect the normal operation of the network, and even lead to huge economic losses. In order to be more accurate and efficient in the traffic detection, in this paper, we propose an N-ARMA based traffic anomaly detection model. We also conduct extensive experiments to verify the higher accurate ratio and recall ratio of our model by comparing with other traffic anomaly detection methods.

Keywords*: Time Series Data; Network Traffic; Anomaly Detection; N-ARMA; SAS*

1. Introduction

The continuous development of the Internet makes the Internet services and applications much more various, but at the same time, the network attacks also emerge frequently, such as worm invasions, denial of service attacks. They can lead to large-scale network paralysis and huge economic losses. Therefore, effective traffic anomaly detection and warning is crucial.

These days, a lot of time series anomaly detection technologies have been applied in the traffic warning area [1][2] [3], such as wavelet analysis [4][5][6], permutation entropy [7][8][9], support vector machine [10], and decision tree [11]. With the deepening of these researches, many commercial network management systems are launched, and the management system based on the predictive model has been gradually applied. However, the accuracy and breadth of the prediction still need to be improved.

In this paper, in order to improve the accuracy of the traffic anomaly detection, we propose an N-ARMA (Normal distribution based on Auto-Regressive and Moving Average) traffic anomaly detection model, and verify the higher value of its accuracy by comparing with methods based on wavelet analysis and permutation entropy.

The rest of this paper is organized as follows. In Section 2 we provide an overview of the related work; Section 3 gives a brief description of the ARMA (Auto-Regressive and Moving Average) model; our N-ARMA traffic anomaly detection model and the implementation are described in Section 4, whereas the experimental results are presented in Section 5; the last section concludes the paper and gives some directions for future work.

2. Related Work

There have been several efforts towards studying the time series anomaly detection, especially about the traffic anomaly. Sun *et al.* [4] applied the two-dimensional diffusion wavelets (DW) transform in traffic matrix analysis and anomaly detection, and the results

[*] Corresponding Author

ISSN: 2005-4270 IJDTA

shows the effectiveness of DW based technique in traffic matrix analysis and anomaly detection in practical networks

Grane and Veiga [5] focused on anomaly detection and correction in the financial data and proposed a general detection and correction method based on wavelets. The effectiveness of the new proposal was tested by an intensive Monte Carlo study for six well-known volatility models.

Since the entropy is a powerful tool for the analysis of time series and it allows describing the probability distributions of the possible state of a system. According to the idea of calculating entropy based on permutation patterns for the understanding of complex and chaotic systems, Zanin *et al.* [8] analyzed the theoretical foundations of the permutation entropy.

Zhou *et al.* [9] provided a visual analytic tool in their work, besides calculating entropy-based traffic metrics. The tool also can provide coherent visual analysis that makes entropy-based traffic features more intuitive and helps users interpret network data and more quickly identify traffic anomalies.

Apart from the above, based on SVM, Catania *et al.* [10] presented an approach for autonomous labeling of normal traffic as a way of dealing with situations where class distribution does not present the imbalance required for SVM algorithm. Experiments show that the use of the proposed approach outperforms existing SVM alternatives.

By using the decision tree learning, Ignase *et al.* [11] proposed a novel scheme and built a system to detect and classify anomalies that are based on an elegant combination of frequent item-set mining with decision tree learning. They achieved good overall classification accuracy and a false-positive rate after evaluating their scheme using traffic traces from two real networks.

Also, the analysis of the ARMA model has been studied extensively and achieves a lot of results. Zheng *et al.* [12] extended the generalized auto-regressive moving average (GARMA) model in such a way that the resulting ARMA model in the transformed space has a martingale difference sequence as its error sequence. Huang [13] developed an ARMA modeling method using a robust Kalman filtering. Boularouk *et al.* [14] and Bhattacharya [15] *et al.* presented a new approach for the optimization of ARMA model's parameters' estimation in case of different order respectively.

3. ARMA Model

ARMA [16] model is an important method in the field of time series analysis, which is based on the Auto-Regressive (AR) model and the Moving Average (MA) model.

AR model uses the potential patterns of the historical data to predict how the variable changes in future, and its formula is as Equation (1).

$$X_t = \emptyset_1 X_{t-1} + \cdots + \emptyset_p X_{t-p} + \varepsilon_t \quad (1)$$

The underlying assumptions of AR model are: X_t is only directly related to $X_{t-1}, \ldots, X_{t-p}$ but independent of X_{t-j} $(j = p+1, p+2, \ldots)$; Furthermore, ε_t is a white noise sequence, has a mean of 0.

By using the simple smoothing forecasting techniques, ARMA can be used to predict the long term trend of the time series. Its basic idea is: according to the historical time series data, we can predict the long-term trend by calculating the average of a certain amount time series data in turn. Usually, we can use $\mathrm{ARMA}(p, q)$ to represent the ARMA model, which consist of an $\mathrm{AR}(p)$ and an $\mathrm{MA}(q)$, and its formula is as Equation (2).

$$X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q} \quad (2)$$

ARMA model's underlying assumptions are: if X_t is only related to X_{t-m} $(1 \le m \le p)$ and ε_{t-n} $(1 \le n \le q)$ but independent of $X_{t-j} (j = p+1, p+2, \ldots)$ and ε_{t-k} $(k = q+1, q+2, \ldots)$, then ε_t must be independent of $\varepsilon_{t-i} (i = q+1, q+2, \ldots)$ and X_{t-j} $(j = p+1, p+2, \ldots)$.

4. N-ARMA Traffic Detection Model

4.1. N-ARMA Model Introduction

N-ARMA traffic anomaly detection model is based on the ARMA prediction model. By predicting the deviation sequence from the historical data, and then fitting the deviation sequence with the normal distribution, we can get the confidence intervals. The N and the ARMA represent the normal distribution and the auto-regressive and moving average model respectively.

The network traffic characteristic is one of the important factors in network traffic model research. Because of the complexity and diversity of the network structure and the frequently sudden behavior of the network, traditional models such as Poisson model and Markov model are no longer suitable for the Internet traffic analysis and prediction.

Numerous studies have shown that the actual network traffic has self-similarity, that is to say in the statistical sense, the actual network traffic displays a self-similarity and growth [17][18][19], and this is an important basis for the proposing of the N-ARMA model in this paper. However, the network traffic also shows an obvious periodicity, for example, there shows the similarity in the network traffic trends every day or every week. While the N-ARMA model is suitable for the stable sequence without white noises, we need to stabilize the traffic sequence before using the N-ARMA model.

4.2. N-ARMA Model Building

The formula of the N-ARMA traffic anomaly detection model is as follows,

$$\begin{cases} X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q} \\ \left(\hat{x}_t(l) \mp 1.96\sqrt{\sigma_\varepsilon{}^2}\right) \end{cases} \tag{3}$$

where the $\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q, \mu, \sigma_\varepsilon^2$ are parameters to be estimated, the equation $X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q}$ is the auto-regressive moving average equation, equation $\left(\hat{x}_t(l) \mp 1.96\sqrt{\sigma_\varepsilon{}^2}\right)$ is the anomaly detection confidence interval.

In the next part, we will introduce the ARMA equation **F1** and anomaly detection confidence interval equation **F2**.

$$F1: \quad X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q} \tag{4}$$

$$F2: \quad \left(\hat{x}_t(l) \mp 1.96\sqrt{\sigma_\varepsilon{}^2}\right) \tag{5}$$

4.1.1. F1 (Auto-Regressive Moving Average Equation)

The equation's building mainly includes five processes: stabilizing the traffic sequence, determining the model's order, estimating the parameter, checking the model and optimizing the model.

We use the analysis of variance (ANOVA) [20] to stabilize the traffic sequence. Specifically, using the S_{ij} to represent the ith 5min observed value of jth day in one week, where $j = 1,2,3,4,5;\ i = 1,2,3,\ldots,288$. With the increasing of the average observed value, the variance also increases, but such sequences cannot be fitted by using standard normal distribution. Therefore, we take log of the traffic sequence firstly, and then divide the $\ln S_{ij}$ into four parts, as shown in Equation(6),

$$\ln S_{ij} = \mu + \alpha_i + \beta_j + y_{ij} \tag{6}$$

where μ represents the average value of this week, α_i represents the difference between the ith average value and the μ, β_j is the difference between the average value of jth day and μ, y_{ij} is the residual, and all the above should satisfy: $\sum_i \alpha_i = 0,\ \sum_j \beta_j = 0$.

After this process, we can eliminate factor "different days in one week" and factor "different times in one day" from the original observed value S_{ij}. Obviously, we can also eliminate other potential factors according to different needs.

Determining the model's order is actually estimating the p and q of the model according to the ACF coefficient (autocorrelation coefficient) and the PACF coefficient (partial autocorrelation coefficients). The fundamental principles are shown in Table 1, where $\hat{\rho}_k$ and $\hat{\phi}_{kk}$ are ACF coefficient and PACF coefficient respectively.

Table 1. Principles in Determining Model's Order

$\hat{\rho}_k$	$\hat{\phi}_{kk}$	Model's Order
trailing	$\hat{p}$ order truncation	p=$\hat{p}$, q=0
$\hat{q}$ order truncation	trailing	p=0, q=$\hat{q}$
trailing	trailing	p=$\hat{p}$, q=$\hat{q}$

In the next step, we need to estimate the unknown parameters by using the current observed value. There are $p+q+1$ unknown parameters in our model: $\phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q, \mu$.

Parameter μ is the average value of the current sequence, we can estimate it by using moment estimation, which estimates the ensemble average with sample average,

$$\hat{\mu} = \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{7}$$

After estimating μ, the number of unknown parameters has been reduced to $p+q$, and we estimate them by using the least square method.

Model checking mainly includes the significance test and the parameter test.

Through the significance test of the model, we can judge whether it can effectively extract the relevant information from all the samples. The test statistic is defined as LB,

$$LB = n(n+2)\sum_{k=1}^{m}\left(\frac{\hat{\rho}_k^2}{n-k}\right) \sim \chi^2(m), \forall m > 0 \tag{8}$$

If the hypothesis is rejected, it indicates that some relevant information still remain in the residual sequence, and the fitting model is not significant; otherwise, it is significant and effective.

The main purpose of the parameter test is to exclude some inessential parameters, simplify the model and make the model easier to use and analyze. We judge the parameters mainly according to whether the parameter is significantly zero, which means that this parameter does not have significant impact on the dependent variable, so we can delete it from the model.

The purpose of model optimization is to select the optimal one from the candidates. In the scope of certain requirements, there are usually multiple models which can go through the significance test, parameter test and effectively fit the trend of observed sequence. Therefore, we need a selection strategy to choose the optimal model. In this paper, we use the AIC [21] criterion and the SBC [22] criterion to do the optimal selection.

4.1.2. $F2$ (Anomaly Detection Confidence Interval Equation)

We assume that the predicted value of traffic at time t is $\hat{x}_t$, while the actual value at that time is x_t. By calculating the difference between the predicted value and the actual value, we can know whether the current traffic value is consistent with the traffic trend. However, because of the periodicity and randomness of the traffic, it is almost impossible that the predicted value will be exactly the same with the actual one. Therefore, when the difference between the predicted value and the actual value is within a credible range, we think the current traffic value is normal; otherwise it is an outlier.

Since the ε_t displays a randomness and fluctuating around zero, it can be considered that ε_t obeys to normal distribution with the mean of 0. In this paper, we use normal distribution to fit our data.

According to the principle of 3σ in normal distribution, at least 95 percent of all the traffics should be the normal traffic, therefore we choose ∓ 1.96 here as the effective range of normal distribution, that is, $\left(\hat{x}_t(l) \mp 1.96\sqrt{\sigma_\varepsilon{}^2}\right)$, where the parameter σ can be calculated by the predicted deviation sequence of historical data.

5. Experiments and Analysis

5.1. Data Preprocessing

In this paper, we use the caching workload data of the online services for experiments. As shown in Figure 1, we select the average flow rate per five minutes as sample points, so, we have 288 points per day. The horizontal axis represents the time points, and the vertical axis represents the flow rate in b/s.

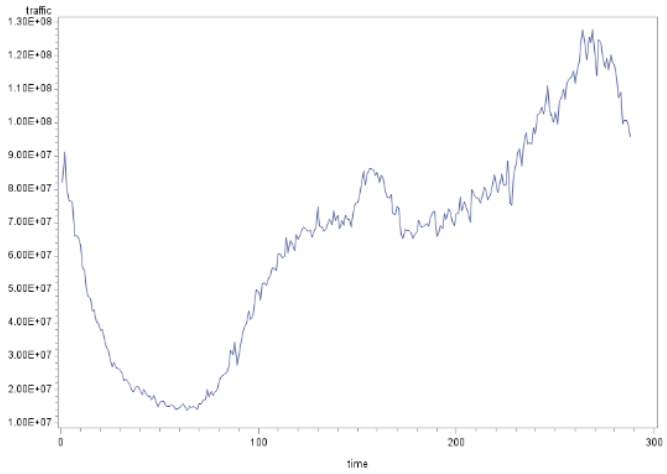

Figure 1. Caching Workload

Further, for each point we use a logarithmic scale for the flow rate and get a more readable and smooth curve, which is as follows.

5.2. N-ARMA Model Building

Before building the model, we firstly calculate the preprocessed traffic sequences' autocorrelation coefficients and partial autocorrelation coefficients, and then do the stationary test and randomness test. If the sequence is not stable or is just the white noise sequence, we need to reconsider the preprocessing of the traffic sequence; if it is stable, we can get the p and q of $F1$ model according to ACF and PACF coefficients.

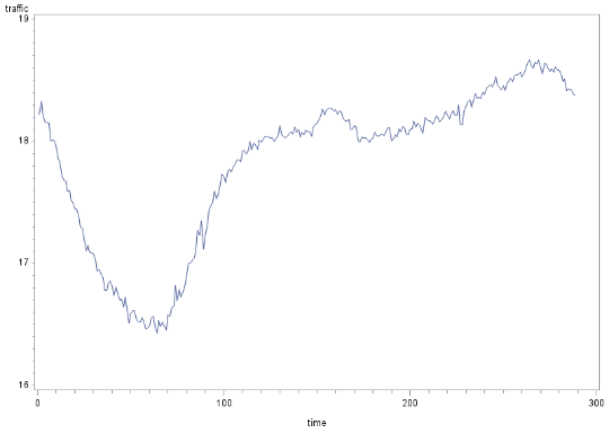

Figure 2. Logarithmic Caching Workload

In this paper, we use the ARIMA's statement "IDENTIFY" in SAS to do the stationary test and randomness test.

Table 2 shows the result of white noise's autocorrelation check. As shown in the table, p-value ($Pr > ChiSq$) is smaller than 0.0001 when the Lag is 6, and the result indicates that the current sequence is not a white noise sequence.

Table 2. Autocorrelation Check for White Noise

To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	1692.61	6	<.0001	0.994	0.989	0.984	0.979	0.972	0.964

Figure 3 shows the result of traffic's trend and correlation analysis. The ACF part indicates that the current sequence is smearing, in another word, the sequence cannot be attributed to zero quickly. Therefore, the current sequence is not stable.

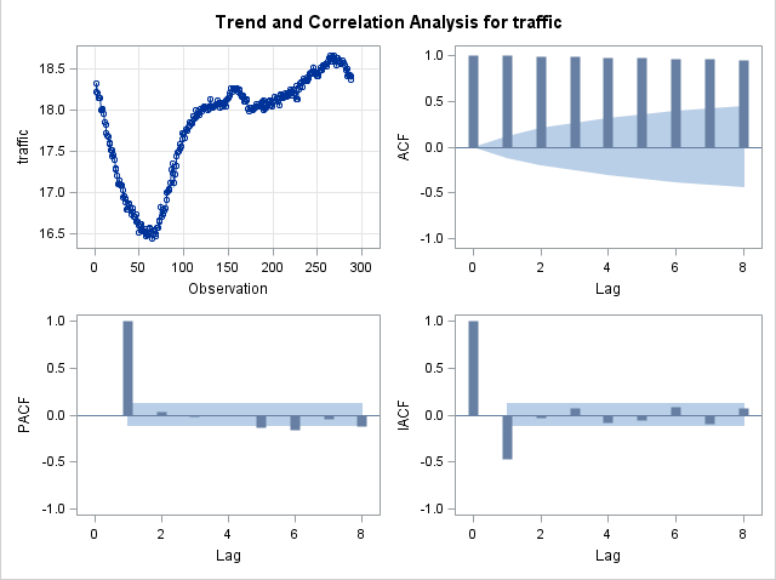

Figure 3. Trend and Correlation Analysis for Traffic

Through the above analysis, we can know the current sequence is neither a white noise sequence nor a stable one, so the F1 model cannot be built here. Therefore, we need to stabilize the current sequence by using the first-order difference method.

Table 3 provides the autocorrelation check result of first-order difference processed white noise. It can be seen from the table that the p-value ($Pr > ChiSq$) is smaller than 0.0001 when the Lag is 6, the result indicates that the current sequence is not a white noise sequence.

Table 3. Autocorrelation Check for White Noise (First-order Difference Processed)

To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	31.23	6	<.0001	-0.131	0.072	0.025	0.205	0.161	0.125

Figure 4 provides the trend and correlation analysis of first-order difference processed traffic. The ACF part indicates that the current sequence can be attributed to zero quickly when the Lag is 1, so, the current sequence is stable.

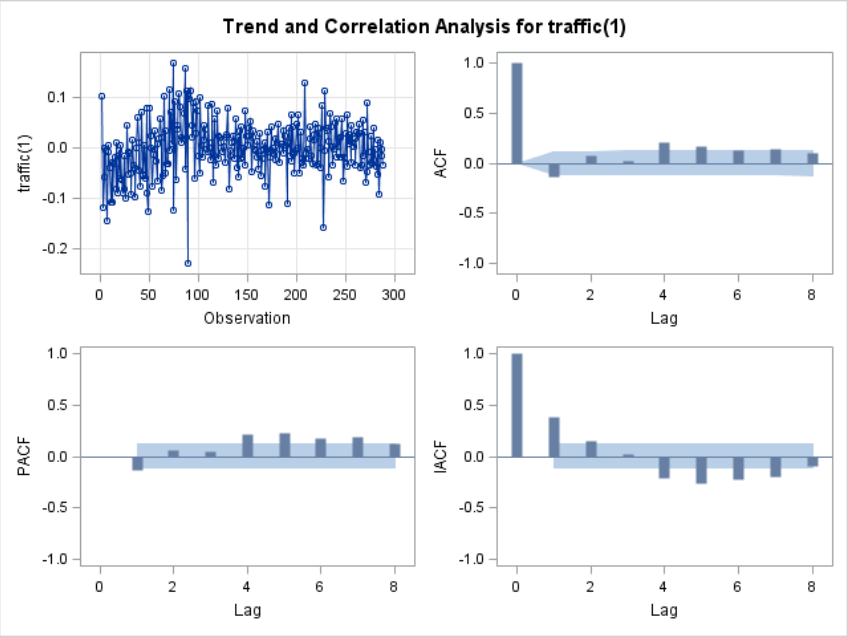

Figure 4. Trend and Correlation Analysis for Traffic (First-order Difference Processed)

After the first-order difference processing, the current sequence is stable and not a white noise sequence now, therefore, we can build the F1 model. According to Figure 4, when the Lag is 1, the ACF of the sequence can be attributed to zero quickly, and we would use the $F1(1, q)$ model to fit the current sequence in the next step.

In SAS, we use the "ESTIMATE" statement of ARIMA to do the model parameters estimation and model checking. In this step, we verify four models from $F1(1, 0)$ to $F1(1, 3)$ and select two candidates which are $F1(1, 2)$ and $F1(1, 3)$ in the model checking process for further optimization.

Here, we select the optimal model according to the AIC criterion and the SBC criterion. Table 4 shows the fitting result of $F1(1, 2)$ and $F1(1, 3)$. According to the contents of the criterion, the smaller the value of AIC and SBC is, the better the model fits. Obviously, model $F1(1, 3)$ has a better fitting than $F1(1, 2)$.

Table 4. $F1(1,2)$(Left) and $F1(1,3)$(Right)

$F1(1,2)$		$F1(1,3)$	
Constant Estimate	-0.00014	Constant Estimate	-0.00022
Variance Estimate	0.002435	Variance Estimate	0.00236
Std Error Estimate	0.049345	Std Error Estimate	0.048583
AIC	-908.675	AIC	-916.627
SBC	-894.037	SBC	-898.329
Number of Residuals	287	Number of Residuals	287

Figure 5 provides the fitting result of $F1(1,3)$. The green line is the real data, and the red line is the predicted one.

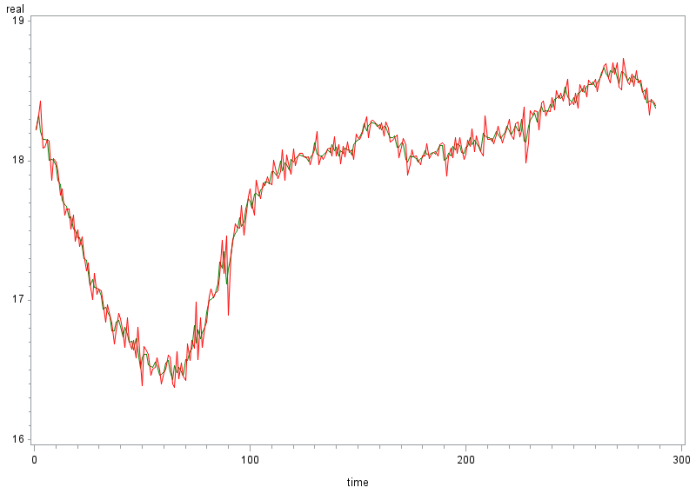

Figure 5. Fitting Result of $F1(1,3)$

Finally, we build the $F2$ anomaly detection confidence interval equation. By using the sample data, we build a normal distribution model and get the deviation distribution $\varepsilon_t \sim N(0,0.08^2)$.

According to the $F2$ formula, we get the confidence interval $(\hat{x}_t(1) - 0.1568,\ \hat{x}_t(1) + 0.1568)$. Therefore, we can detect the outlier by verify whether or not its real value is in the interval of the predicted one.

5.3. Outlier Detection Experiments

In the outlier detection experiments, we compare our N-ARMA method with the wavelet analysis and the permutation entropy. The four datasets we used in our experiments are as follows:

S1: 5 minutes granularity of traffic data for one week, 2016 points, 39 outliers.
S2: 5 minutes granularity of traffic data for one month, 8640 points, 133 outliers.
S3: 60 minutes granularity of traffic data for one month, 720 points, 11 outliers.
S4: 60 minutes granularity of traffic data for half year, 4320 points, 34 outliers.

Table 5. Experimental Results of Wavelet Analysis, Permutation Entropy and N-ARMA

Dataset	Algorithm	Number of Outliers Detected	Number of Correct Outliers
S1	Wavelet Analysis	40	33
	Permutation Entropy	46	31
	N-ARMA	40	36
S2	Wavelet Analysis	125	115
	Permutation Entropy	139	113
	N-ARMA	132	121
S3	Wavelet Analysis	11	9
	Permutation Entropy	14	9
	N-ARMA	10	10
S4	Wavelet Analysis	33	28
	Permutation Entropy	37	29
	N-ARMA	32	31

The experimental results are presented in Table 5.

Table 6 gives the experimental result of three methods' recall ratio and accurate ratio. The result indicates that N-ARMA model has a better performance (all above 90%) over the wavelet analysis and the permutation entropy in recall ratio and accurate ratio. On the other hand, the N-ARMA model is relatively easier to be implemented. In sum, the N-ARMA model proposed in this paper is significantly effective in traffic anomaly detection.

Table 6. Accurate Ratio and Recall Ratio of Wavelet Analysis, Permutation Entropy, and N-ARMA

Dataset	Wavelet Analysis		Permutation Entropy		N-ARMA	
	Accurate Ratio	Recall Ratio	Accurate Ratio	Recall Ratio	Accurate Ratio	Recall Ratio
S1	82.5000	84.6154	67.3913	79.4872	90.0000	92.3077
S2	92.0000	86.4661	81.2950	84.9624	91.6667	90.9774
S3	81.8182	81.8182	64.2857	81.8182	100.000	90.9091
S4	84.8485	82.3529	78.3784	85.2941	96.8750	91.1765
Average	85.2917	83.8132	72.8376	82.8904	94.6354	91.3427

6. Conclusions

In this paper, we propose the N-ARMA traffic anomaly detection model and conduct extensive experiments. The results show that our model has a better performance in the accurate ratio and recall ratio than the other two models, so it is feasible to apply it to the traffic anomaly detection. Since this is just our first attempt to the research on traffic anomaly detection, there still needs more effort to improve it.

In terms of the accuracy, we can combine our N-ARMA model with the learning algorithm, which can be used for outlier judgment. Also, we can do a more effective preprocessing of the historical data to further improve the accuracy of the anomaly detection.

In specific applications, we need to further study the performance of our model in real network, especially in the big data environment.

Acknowledgments

This research is supported by the Science and Technology Key Project of Fujian Province, China (2014H0044); Science and Technology Guiding Project of Fujian Province, China (2015H0037, 2016H0035); Science and Technology Project of Xiamen, China (3502Z20153026).

References

[1] P. J. Brockwell and R. A. Davis, "Time series: theory and methods", Springer Science & Business Media, **(2013)**.
[2] A. Masanao, "State space modeling of time series", Springer Science & Business Media, **(2013)**.
[3] M. H. Hu, S. T. Tu, F. Z. Xuan and Z. D. Wang, "On-Line Structural Damage Feature Extraction Based on Autoregressive Statistical Pattern of Time Series", ASME 2014 Pressure Vessels and Piping Conference, American Society of Mechanical Engineers, Sunnyvale, California, USA, **(2014)**.
[4] T. Sun, H. Tian and X. Mei, "Anomaly detection and localization by diffusion wavelet-based analysis on traffic matrix", Computer Science and Information Systems, **(2015)**, pp.59-59.
[5] A. Grané and H. Veiga, "Wavelet-based detection of outliers in financial time series", Computational Statistics & Data Analysis, vol. 54, no. 11, **(2010)**, pp. 2580-2593.
[6] L. Wei and A. A. Ghorbani, "Network anomaly detection based on wavelet analysis", EURASIP Journal on Advances in Signal Processing, **(2009)**, pp. 4.
[7] C. Bandt and B. Pompe, "Permutation entropy: a natural complexity measure for time series", Physical review letters, vol. 88, no. 17, **(2002)**, pp. 174102.
[8] M. Zanin, L. Zunino, O. A. Rosso and D. Papo, "Permutation entropy and its main biomedical and econophysics applications: a review", Entropy, vol. 14, no. 8, **(2012)**, pp. 1553-1577.
[9] F. Zhou, W. Huang, Y. Zhao, Y. Shi, X. Liang and X. Fan, "ENTVis: A Visual Analytic Tool for Entropy-Based Network Traffic Anomaly Detection", Computer Graphics and Applications, IEEE, vol. 35, no. 6, **(2015)**, pp. 42-50.
[10] C. A. Catania, F. Bromberg and C. G. Garino, "An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection", Expert Systems with Applications, vol.39, no. 2, **(2012)**, pp.1822-1829.
[11] I. P. Oliva, I. C. Uroz, P. B. Ros, X. Dimitropoulos and J. S. Pareta, "Practical anomaly detection based on classifying frequent traffic patterns", Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on. IEEE, Orlando, Florida, USA, **(2012)**.
[12] Z. Tingguo, H. Xiao and R. Chen, "Generalized ARMA models with martingale difference errors", Journal of Econometrics, vol. 189, no. 2, **(2015)**, pp. 492-506.
[13] H. Lei, "Auto Regressive Moving Average (ARMA) Modeling Method for Gyro Random Noise Using a Robust Kalman Filter", Sensors, vol. 15, no. 10, **(2015)**, pp. 25277-25286.
[14] Y. Boularouk and K. Djeddour, "New approximation for ARMA parameters estimate", Mathematics and Computers in Simulation, vol. 118, **(2015)**, pp. 116-122.
[15] S. S. Roy and S. Bhattacharya, "Asymptotic distribution of the estimated parameters of an ARMA (p, q) process in the presence of explosive roots", APPLICATIONES MATHEMATICAE, vol. 39, no. 3, **(2012)**, pp. 257-272.
[16] C. B. Seon, "ARMA model identification", Springer Science & Business Media, **(2012)**.
[17] J. Beran, R. Sherman, M. S. Taqqu and W. Willinger, "Long-range dependence in variable-bit-rate video traffic", Communications, IEEE Transactions on, vol. 43, no. 2, **(1995)**, pp. 1566-1579.
[18] M. Grossglauser and J. Bolot, "On the relevance of long-range dependence in network traffic", IEEE/ACM Transactions on Networking (TON), vol. 7, no. 5, **(1999)**, pp. 629-640.
[19] M. Ghaderi, "On the relevance of self-similarity in network traffic prediction", School of computer science, university of Waterloo. CS-2003-28, **(2003)**.
[20] R. J. Freund and R. C. Littell, "SAS for linear models: a guide to the ANOVA and GLM procedures", Sas Institute, **(1981)**.
[21] H. Akaike, "Factor analysis and AIC", Psychometrika, vol. 52, no. 3, **(1987)**, pp. 317-332.
[22] S. Yosiyuki, M. Ishiguro and G. Kitagawa, "Akaike information criterion statistics", Dordrecht, The Netherlands: D. Reidel, **(1986)**.

Authors

Pingping Gu, Oct. 11th, 1982, China, Current position, grades: She is currently a lecturer at the Tan Kah Kee College Xiamen University. University studies: She received the B.S. degree and M.S. degree in Computer Science from Xiamen University in 2004 and 2007, respectively. Scientific interest: Software Engineering; Computer Vision; Data Mining. Publications <number or main>: She is the author of more than 10 publications. Experience: In 2009, She was appointed a lecturer at Computer Science Department in the Tan Kah Kee College Xiamen University.

Shijing Zhang, December 27, 1991, China, Current position, grades: Master Student. University studies: She has received the B.S. degree in Software Engineering from Xiamen University in 2014 and would receive M.S. degree in 2017. Scientific interest: Data Mining; Information Management; Web Development Experience: During the postgraduate stage, she served as a developer at Electronic Commerce System Based on J2EE, the International Teaching Quality Assurance System for Xiamen University and one Precise Healthy Customization Platform in Xiamen University.

Zhi Min Huang, October 12, 1988, China, Current position, grades: Master University studies: He received the B.S. degree and M.S. degree in Computer Science from Xiamen University in 2011 and in 2014, respectively. Scientific interest: Software Engineering; Information Management; Big Data Technology Experience: In June 2011, he served as a developer at Digital Media Center of Software School in Xiamen University. Since 2013, he has been CTO of one company.

Qingfeng Wu, September 20, 1977, China, Current position, grades: He is currently a Professor at the Xiamen University. And he is also the Vice Director of the Software Engineering Lab. University studies: He received the B.S. degree in system engineering and the PH.D. degree in Computer Science from Xiamen University in 2000 and 2007, respectively. Scientific interest: Software Engineering; Visual Information Processing; Computer Vision. Publications <number or main>: He is the author of more than 50 peer-reviewed publications and has two patents in computer vision. Experience: In 2009, Prof. Wu was appointed a project leader for research and development at software school in Xiamen University.