Eng Int Syst (2021) 3: 183–189

Engineering
Intelligent Systems

Network Traffic Monitoring and Real-Time Risk Warning Based on Static Baseline Algorithm

Li Fu* and Zhu Jingli

Zhengzhou Preschool Education College, ZhengZhou, HeNan, 450000, China

In this research, firstly the historical data of the sample is analyzed, and the Gaussian process static baseline algorithm is used to study this historical data. The ant colony algorithm is applied to optimize the structure and, finally, the basic value of the Gaussian process static baseline algorithm is realized. The adjustment method achieves real-time monitoring of network traffic and provides early warning of network risks through a series of studies. The research results indicate that the baseline calculation method used in this study produces greater accuracy than the traditional limit calculation method, and can also make better use of the statistical characteristics of noise. Later, we will study the algorithm of data distribution according to time series in a noisy environment, compare this method with other kinds of prediction algorithms, and promote it. In summary, the network traffic monitoring system developed and designed in this study meets the basic daily usage needs, and also meets the index requirements of the cooperative project.

Keywords: Baseline algorithm; Gaussian process; Traffic monitoring; Risk warning

1. INTRODUCTION

The most basic requirement of the existing network traffic monitoring system is that it receives all the data packets flowing into the system, because any data packet may contain Trojan horse viruses, making it an attack packet. However, currently, the expansion of network bandwidth is outpacing the growth rate of system processors. According to Gilder's law, network bandwidth tends to double every six months, and the growth rate of network bandwidth is at least three times faster than the growth rate of computer processor performance [1]. Key issues associated with the capturing of data packets are: how to ensure that in an ever-expanding network environment, that is, in a scenario where device hardware resources are limited, network data packets are captured with minimal packet loss. This research proposes a method for the real-time monitoring of network traffic based on the automatic adjustment of high-speed process technology with a static baseline algorithm, which means that it satisfies the needs of large server clusters for computer processor performance warning settings and signal shielding. This is through an analysis of the statistical characteristics of historical sample data, combined with the distribution of historical sample data to make accurate predictions based on sample data [2]. This method first collects the noise of the historical data of the sample, and then combines the ant colony algorithm to create an automatic adjustment method of the Gaussian process and finally calculates the static baseline. The research results indicate that compared with other calculation methods, this algorithm not only guarantees higher accuracy, but can also improve the efficiency of both the computer calculation and the system's early warning of network risks.

*Address for correspondence: Li Fu, Zhengzhou Preschool Education College, ZhengZhou, HeNan, 450000, China, Email: lifu9987456@163.com

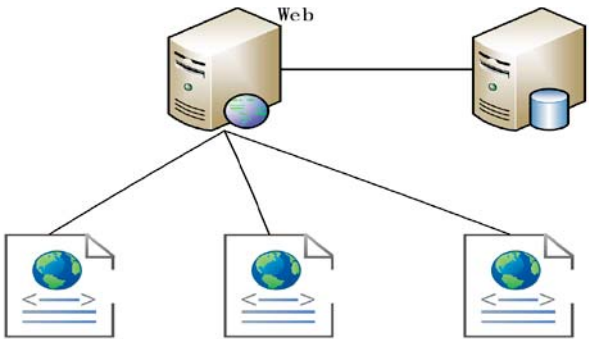

Figure 1 B/S mode diagram.

2. THEORETICAL BASIS OF STATIC BASELINE ALGORITHM

2.1 WEB Theory and Technology

The software development framework consists of three major components: client, server mode and browser (B/S), server mode [3]. The client usually needs to install the corresponding client, and the latter's user interface display and operation are all done through the browser, as shown in Figure 1.

2.2 Static Baseline Algorithm Based on Adaptive Gaussian Process

In this study, the researchers designed a set of active monitoring algorithms to meet the requirements of computer performance warning settings and signal shielding. This algorithm structure includes a "softswitch network performance real-time prediction algorithm" and "active monitoring of computer performance" Three parts: real-time warning algorithm" and "active softswitch network optimization algorithm" [4]. This algorithm is used to collect a large number of network performance early warning programs and basic data about the corresponding equipment. The function of the algorithm is to actively monitor computer performance and to optimize the network early warning signal in real time. The "real-time early warning algorithm for active monitoring of computer performance" consists of: "a static baseline algorithm", "a tolerance calculation method", and "a network early warning generation mechanism" [5].

After the static baseline is obtained by this algorithm, an appropriate tolerance value is selected to float up and down the static baseline value to obtain the tolerance line, and the value represented by the line is used as the limit value for the subsequent network signal warning mechanism. When the detected data exceeds this limit value, the corresponding network warning signal is generated according to the characteristics of the computer performance index.

The Gaussian process algorithm is actually a machine learning algorithm with an implementation monitoring mechanism, so it is commonly used in the field of machine learning. This algorithm is based on the Bayesian framework to calculate nonlinear network prediction problems [6]. The Gaussian process is formed by multiple sets of random variables. The combination of any variable in the set obeys the joint Gaussian distribution, and its specific values are determined by the mean function and the covariance function. Compared with support vector machines and neural network methods, this calculation method is a non-parametric probability model. It predicts the input value during the calculation process and also obtains the probability estimate of this prediction [7]. In the prediction model, there are fewer unknown parameters, which is convenient for subsequent optimization and easier to converge. First, the Gaussian process regression prediction model presets the input target value y which is different from the true value t:

$$y = t + \varepsilon \tag{1}$$

The prior distribution of the input target value y is:

$$y \sim N\left(0, K + \sigma_n^2 I\right) \tag{2}$$

The joint Gaussian prior distribution formed by the output y of n training samples and the output y of 1 test sample is:

$$\begin{bmatrix} y \\ y^* \end{bmatrix} \sim N\left\{0, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,x^*) \\ K(X,x^*)^T & k(X^*,X^*) \end{bmatrix}\right\} \tag{3}$$

$$L = \ln p(y \mid X)$$
$$= -\frac{1}{2} y^T \left(K + \sigma_n^2 I\right)^{-1} y - \frac{1}{2} \ln \left|K + \sigma_n^2 I\right| - \frac{n}{2} \ln 2\pi \tag{4}$$

When the optimal parameters are obtained, the prediction can commence. The specific process is: predict the most likely output value corresponding to x on the basis of the training set according to the Bayesian principle [8]. The Bayesian principle is adopted in order to update the probability forecast distribution in real time with the help of observation data.

$$y^* \mid x^*, X, y \sim N\hat{y}(x^*), \sigma(x^*) \tag{5}$$

y mean and variance are

$$\hat{y}(x^*) = k^T(x^*)(K + \sigma_n^2 I)^{-1} y \tag{6}$$

$$\sigma(x^*) = k(x^*, x^*) - k^T(x^*)(K + \sigma_n^2 I)^{-1} k(x^*) \tag{7}$$

The steps are: collect one month's historical data, with a daily data set size of 1440. Use the previous day's data minus the next day's data to obtain new data. Perform normal distribution analysis on the new data to obtain the results

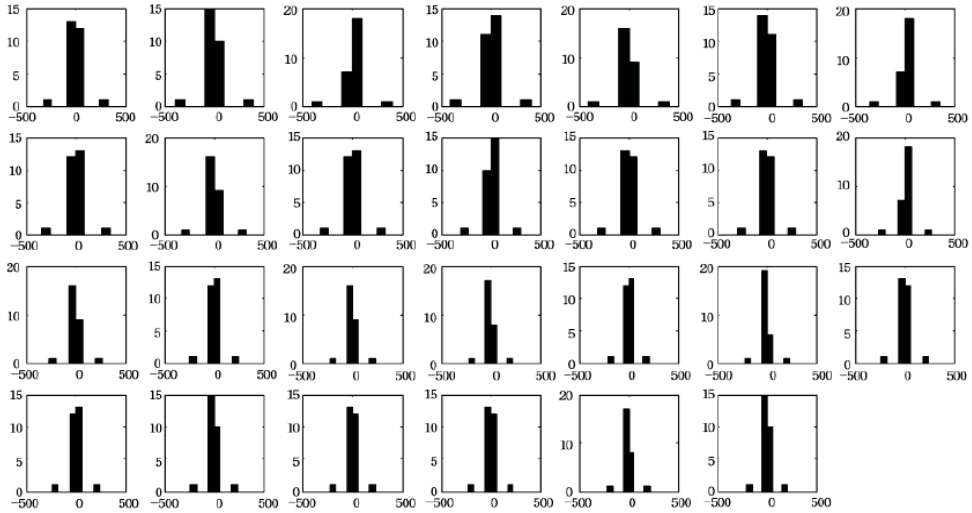

Figure 2 Analysis of normal distribution of sample noise.

Table 1 Parameter estimation of normal distribution of sample noise.

Sequence	Mean	Variance	Sequence	Mean	Variance
1	−0.00926	89.8136	15	0.015926	71.004
2	−0.01037	109.5037	16	−0.01556	67.63911
3	−0.05259	111.2537	17	−0.01926	63.60109
4	−0.03889	114.051	18	0.005556	61.2816
5	0.05667	113.5536	19	0.0363	56.07392
6	−0.04037	101.2621	20	0.031481	61.55756
7	0.03144	97.63631	21	0.00711	65.87238
8	−0.01778	95.02861	22	0.041481	65.30578
9	0.06778	91.03038	23	0.012963	61.86611
10	−0.06	89.94092	24	−0.01926	62.28233
11	−0.01185	84.50101	25	−0.02037	62.06943
12	0.03963	86.82521	26	−0.03071	59.82396
13	−0.0137	80.09664	27	−0.0137	56.75401
14	0.01593	73.61197			

shown in Figure 2 and Table 1. It can be clearly seen that the data basically conforms to the Gaussian distribution, which verifies the Gaussian distribution characteristics of the sample to a certain extent.

3. DESIGN AND IMPLEMENTATION OF ONLINE NETWORK TRAFFIC MONITORING AND EARLY WARNING SYSTEM

3.1 System Requirements Analysis

System requirements analysis is the most important step in the software design process as it can anticipate potential problems and produce a better understanding of the end-user's needs. It is the basis of system design. Only by thoroughly understanding the actual needs of users can we grasp the overall idea and plan of system design [9]. This system monitors basic flow as well as intelligent acceleration flow based on related requirements. After system research and in-depth exchanges, we hope that we can grasp the usage of network bandwidth through regular network-specific traffic analysis, which can be used for timely warning and diagnosis of link congestion and equipment failure. Through observation and analysis of application traffic and corporate user online behavior, irresponsible users who wantonly disseminate offensive and illegal information can be monitored and warned. Real-time interactive data analysis can be used to predict the future development direction of network services, which assists with the planning of an efficient network and ensures that users have a high-quality online experience [10]. Therefore, the online network traffic monitoring system should have the following core functions:

① Real-time network traffic collection, using reasonable implementation methods to realize real-time capture of enterprise import and export network data packets, data packet analysis, data packet information extraction, data packet storage and other functions, and store data packets from the network card to the hard disk.

② Network real-time uplink and downlink rate monitoring, can use technical means to realize real-time query and display of the uplink and downlink traffic rate on a certain network card, query the historical traffic rate for a certain

Table 2 Hardware device table.

Hardware equipment list	
CPU	Intel Atom
RAM	SG
Network card	Intel e1000e
Hard disk	IT

Table 3 Software platform.

Software platform	
System	64-bit Ubuntu 15.10
Development language	C/Python/Html/css/js/PHP
Development tools	QT/Spvder/Notepad

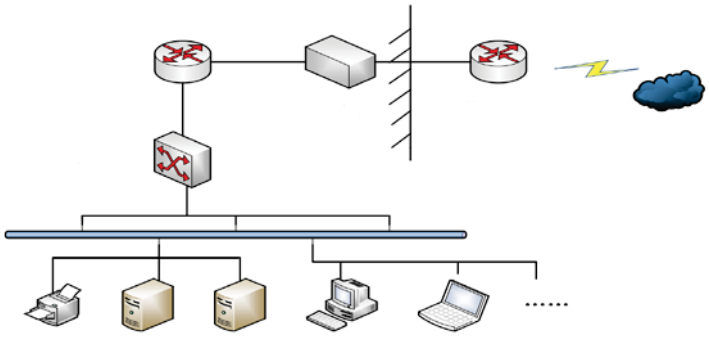

Figure 3 Flow monitoring equipment connection topology diagram.

period of time, and understand the daily use of network bandwidth [11].

③ Query the statistical analysis data of network traffic by IP and Session, complete the analysis of the Internet data of internal users of the enterprise, and understand the traffic usage of the internal network host in order to rationally plan the network and create a safe network environment.

④ Statistics on user access to domain names, complete real-time traffic usage of all domain names accessed by users, and understand the Internet habits of users within the enterprise to detect and promptly curb the spread of viruses.

⑤ Network traffic classification query, complete the classification and identification of enterprise traffic, fully judge the network development trend, and discover network problems such as equipment failures in time through historical data analysis [12]. The basic technical indicators of this system design are as follows: In order to ensure the normal operation of the subsequent functions, the data packet acquisition module is designed to occupy less than 20% of the CPU, less than 10% of the memory, the accuracy of the network traffic query time is 1 min, and the traffic log query time is 7 days. The visualization platform integrates functions such as accessing domain names and addresses, as shown in Table 2.

Table 3 shows the hardware equipment table and software platform. Due to the cost of equipment, the CPU purchased Intel Atom with poor processing performance, and used three Intele1000e network cards, two of which are responsible for monitoring all basic Internet traffic of users, and one is responsible for Monitor the traffic accelerated by the intelligent network system [13]. Due to budget constraints, it is a significantly challenging task to design an effective and efficient traffic monitoring system that meets requirements. The entire system is trialed at the network outlet of Beijing Secoo Trading Co., Ltd. to monitor the online network traffic in real time and determine the daily network usage of the company's internal hosts. This provides a basis for subsequent traffic acceleration strategies and company network management.

3.2 Network Deployment Method

The system equipment is directly bridged at the network exit. Figure 3 depicts the network topology which ensures that all import and export data packets within the company flow through the flow monitoring system.

In Figure 3, the online network monitoring equipment is connected in series with the traffic egress router of the enterprise network port to ensure that all traffic at the network egress can pass through the network monitoring system [14]. Printing and transmission equipment, a database server, Web server, office host, mobile equipment, etc. are connected to the corporate intranet, which is connected to the export router through the upper network switch. The number of internal users is appropriate for its main business scope which includes: publication retail, online retail, photographic equipment, watches, luggage, jewelry, automobiles, toys,

Table 4 Test software environment.

Software platform	Information
System	64-bit Ubuntu 15.10
Kernel version	C/Python/ Html/css/js/PHP
PF_RING	version 4.7.0
GCC	version 5.2.1
Python	version 3.6
Dnsmasq	version 2.75
liglittpd	version 1.4.35
php	5.6 U-lubuntii3 4

Table 5 Test set experiment results.

Label	Precision	Recall	fl-score	Sample size
BROWSING	0.97	0.99	0.98	2973
CHAT	0.94	0.91	0.93	792
FT	0.91	0.96	0.93	1182
MAIL	0.95	O.88	0.91	436
P2P	0.99	0.99	0.99	1188
STREAMING	0.94	0.85	0.89	397
VOIP	1.00	0.99	0.99	1915

pet supplies; photography and video services, goods import and export, technology import and export Internet service provision, catering services, etc. [15]. The company's network business is diversified, the upstream and downstream traffic bandwidth is large, and it can collect a wealth of Internet traffic information, providing diverse results for system experiments and tests.

3.3 Test Environment Design

The software testing and development environment used for the design of this system is shown in Table 4, which includes basic information such as system platform version, kernel version, software Lighttpd, Dnsmasq version.

The system equipment was trialed at a company's network outlet which had two outlet links with a bandwidth of 300M. The system equipment collects and analyses in real time all the traffic passing through the network outlet router.

3.4 Analysis of Test Results

The model test results of the test set divided in the ISCX data set are shown in Table 5. The first column contains the classification labels of seven types of samples, the second column shows precision, the third column shows recall, and the fourth column is the f1_score. The five columns indicate the number of samples in the test set [16]. Table 5 shows that the number of test samples in the entire test set is about 9,000 network data stream samples. In the test results, the overall precision, recall, f1_score, and scores are all high, and the algorithm model performs well with the data set. Among the samples of each category, the precision and recall of VOIP, BROWSING and P2P samples are close to 0.99. These three types of samples account for a large proportion of the data, and the final classification effect is relatively better.

The experiment counts the traffic data on the acceleration network card from 9:00 to 12:00 on March 19, 2019. The traffic recognition module uses the trained stacking model to classify the flow data. The stream data processed by the model can be directly input into the classification model after preprocessing is completed [17].

3.5 Early Warning Accuracy Analysis

Under normal circumstances, the abnormal traffic is marked as follows: first learn the historical traffic distribution, which may require some understanding of the future traffic distribution trend, and then observe whether the current traffic value deviates greatly from what is expected. The algorithm proposed in this paper is also based on this [18]. The prediction of the model is regarded as a summary of the future distribution flow expected by the staff based on the historical flow data, and it is judged whether the deviation of the current observation point during a certain period of time is similar to other. The points are quite different (the greater the difference means the lower the density, that is, the greater the outlier factor), and the deviation exceeds the threshold and it is judged as an abnormal flow value. The whole detection process of the algorithm follows the steps of manual detection performed by the staff, so the results are more consistent with those obtained through manual detection.

In order to eliminate the influence of individual cases, four sets of data were selected for this study. Taking into account that the discriminant criteria for each student is inconsistent, the manually annotation data of the previous 14 days were extracted to learn, and the algorithm parameters were adjusted. The remaining 28 days of data (a total of 8064 observation points) [19] For accuracy comparison analysis, the outliers are marked as the positive class (P), and the normal value is marked as the negative class (N). The final confusion matrix for each set of data, so the accuracy of the positive class sample. This is calculated with:

$$\text{Pr}ecision = \frac{TP}{TP + FP} \tag{8}$$

Table 6 Accuracy of positive samples in each data group.

Data group	Number of positive samples (manually labeled)	Positive sample accuracy rate	Recall rate of positive samples
DI	3	1.000	1.000
D2	11	1.000	0.909
D3	11	0.846	1.000
D4	21	0.875	0.952

The recall rate is calculated with:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

The precision and recall rates corresponding to each group of data are calculated; the results are shown in Table 6 below. When the positive samples (abnormal flow value) [20] are small, the precision and recall rate are higher, even for the D1 data. 1.000, with the increase in the number of positive samples, its precision rate and recall rate decreased slightly, but both achieved the expected results.

4. CONCLUSION

Firstly, in this research, the historical data of the sample is analyzed. The Gaussian process static baseline algorithm is used to study the historical data of the sample, and the ant colony algorithm is applied to optimize the structure, and finally realizes the basic value of the Gaussian process static baseline algorithm. The adjustment method achieves real-time monitoring of network traffic and early warning of network risks through a series of studies. The research results indicate that the baseline calculation method used in this study achieves greater accuracy than the traditional limit calculation method, and can make better use of the statistical characteristics of noise. In future research, we will study the algorithm of data distribution according to time series in a noisy environment, and compare this method with other kinds of prediction algorithms. In summary, the network traffic monitoring system developed and designed in this study meets basic daily usage needs, as well as the index requirements of the cooperative project. However, there are still some problems worth considering and solving. One is that the classification results of network traffic in the intelligent network traffic classification module cannot be displayed in real time; there is a data delay of about five minutes because when the network bandwidth is large, the number of data packets is huge. The classification model cannot classify all flows quickly. Later, parallel processing can be considered as a means of accelerating the identification rate of the flow. At the same time, the updating of the traffic classification model needs to be done manually. The learning and updating of the model cannot be achieved by using online data. This classification belongs to the classification algorithm with tags. The tag category limits the final classification result as there are many types of Internet traffic. A clustering algorithm can be used to improve the flow identification algorithm. Second, when analysing DNS logs, it is impossible to convert all IPs to domain names because the domain names accessed by users in some DNS logs do not include all destination IP addresses. Hence, the final result will be unmapped The domain name is directly replaced by the IP address.

REFERENCES

1. A. Aizawa. An information-theoretic perspective of tf–idf measures. *Inf Process Manag 39*(2003), 45–65.
2. S. Akhtar, D. Gupta, A. Ekbal, P. Bhattacharyya. Feature selection and ensemble construction: a two-step method for aspect-based sentiment analysis. *Knowl-Based Syst 1251*(2017), 116–135.
3. S. Akhtar, T. Garg, A. Ekbal. Multi-task learning for aspect term extraction and aspect sentiment classification. *Neurocomputing 26*(9) (2020), 573–592.
4. O. Alqaryouti, N. Siyam, AA. Monem, K. Shaalan. Aspect-based sentiment analysis using smart government review data. Appl Comput Inform 23(4) (2019), 1236–1257.
5. X. Bai. Predicting consumer sentiments from online text. Decis Support Syst 15(4) (2011), 732–742.
6. AS. Cerqueira, DD. Ferreira, MV. Ribeiro, CA. Duque. Power quality events recognition using a SVM-based method. Electr Power Syst Res 78(9) (2008), 1546–1552.
7. CC. Chang, CJ. Lin. LIBSVM: a library for support vector machines, *Software 47*(3) (2001), 721–735.
8. JR. Chang, MY. Chen, LS. Chen, WT. Chien. Recognizing important factors of influencing trust in O2O models: an example of OpenTable. *Soft Comput 24*(2020), 7907–7923.
9. P. Chaovalit, L. Zhou. Movie review mining: a comparison between supervised and unsupervised classification approaches. In: *Proceedings of the 38th Hawaii International Conference on System Sciences 46*(9) (2005), 2435–2462.
10. LS. Chen, CT. Su. Using granular computing model to induce scheduling knowledge in dynamic manufacturing environments. *Int J Comput Integr Manuf 21*(5) (2008), 569–583.
11. LS. Chen, CC. Hsu, MC. Chen. Customer segmentation and classification from blogs by using data mining: an example of VOIP phone. *Cybernet Syst 40*(7) (2009), 608–632.
12. LS. Chen, CH. Liu, HJ. Chiu. A neural network-based approach for sentiment classification in the blogosphere. *J Inform 5*(2) (2011), 313–322.
13. N. Chouchani, M. Abed. Enhance sentiment analysis on social networks with social influence analytics. *J Ambient Intell Human Comput 11*(4) (2020), 139–149.
14. C. Cortes, V. Vapnik. Support-vector networks. *Mach Learn 20*(1) (1995), 273–297.
15. K. Dave, S. Lawrence. DM. Pennock, Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: *The 12th WWW 42*(1) (2003), 519–528.
16. L. Denecke, W. Nejdl How valuable is medical social media data? Content analysis of the medical web. *Inf Sci 179* (2009), 1870–1880.

17. FT. Giuntini, MT. Cazzolato. MdJD. dos Reis. A review on recognizing depression in social networks: challenges and opportunities. *J Ambient Intell Human Comput 26*(4) (2020), 192–203.
18. O. Gokalp, E. Tasci, A. Ugur. A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification. *Expert Syst Appl 14*(6) (2020), 151–176.
19. MA. Hassonah, R. Al-Sayyed, A. Rodan, et al An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter. *Knowl Based Syst 19*(2) (2020), 105–153.
20. FH. Khan, U. Qamar, S. Bashir. SWIMS: semi-supervised subjective feature weighting and intelligent model selection for sentiment analysis. *Knowl Based Syst 10015* (2016), 97–111.