A Technique for Prediction of Network Congestion Using Neural Network

Yaminee Sahu
Research Scholar, Department of Computer Science and Engineering, Bhilai Institute of Technology, Durg, India.

Sumit Kumar Sar
Associate Professor, Department of Computer Science and Engineering, Bhilai Institute of Technology, Durg, India.

Abstract – In this paper a new congestion detection technique based on neural network is proposed, which considers the practical status of data loss or packet loss within nodes during data communication. For this purpose, mistreatment learning techniques are employed to predict network congestion issues before they begin impacting the performance of service. The described technique specializes in employing a straight forward feed forward neural network to predict severe congestion in an exceedingly expanding network. We also style and apply a simple management technique for limiting the speed of the sinning source in order that congestion can be avoided. The above technique set the degree for a new gesture of network administrator which can be capable of stopping networking issues in preference to repairing them. Simulation results show the average classification rate for the congestion detection appears to be 99% of the total cases.

Index Terms – Neural Network, Congestion Detection, Neural Network prediction, congestion control.

1. INTRODUCTION

Speed, connectivity and reliability are paid more attention in the growing world of networking, as it is crucial for inter-business operations and for availability of product to customer through electronic commerce in the business field. Everyday lives are also affected by networking as the ways in which people stay connected, communicate and seek information has drastically changed.

As a result, growing number of network users, the focus on maintenance and reliability of the network has increased. For enterprises or individuals that depend on the network connection, occurrence of network problem is equivalent to catastrophe. Network connection drops are not just annoying, but often it leads to loss of revenue for companies and commercial users in the order of thousands, or even millions of dollars.

In all forms of electronic commerce such connection drops have thus become a significant problem. There is need of a system to address this difficulty in order to ensure network availability and efficiency by preventing such costly network breakdowns. The initial step towards this end is to create a system with the intelligence is the recognition of incoming network service difficulties as early as possible. Recognition of problem beforehand and therefore changing network parameters can possibly circumvent the problem.

Many firms nowadays provide a stylish suit of network performance measurement tools and alarm rising system for identification of problem. These tools many however simply provide the network administrator a good amount of an information about the difficulty after it has happened, with the hope that this will help the administrator find the cause of the problem and find an appropriate solution quickly. But detection of problem in advance using previous breakdowns is thought by none. Early detection is excellent, due to the fact that it would allow the administrator or the system itself to save the hassle from ever occurring.

The most important factor in our system would be to detect congestion in network with the help of neural network predictive technique before it consequences in network breakdowns and which also identifies the cause of the congestion. For identifying source of the congestion our scheme applies flow rate limitation to avoid congestion overflowing the router buffer. Design and experiments proposed in this paper focus on congestion control, it is noteworthy; however, that the techniques could also be related to other network troubles. It should also be noted that flow rate limitations can be applied directly to the original flow source if it is inside the domain restricted by our scheme in the form of remedy, or it could be applied to the edge router to the domain to which our scheme is applied. Inside the latter case, the limitations will result in the packets of the limited flow being dropped at the edge router to the field. Falling packets at the entrance edge router of one domain will source the packets to be dropped at the way out router of the nearby domain which will treat such dropping as congestion and then will recognize the source of the flow. As a result, our technique can be applied locally at a domain of the decayed network and their congestion solution will iteratively be mapped to the equivalent edge routers of the middle domain until the cause of the flow is originated and learned of the need to reduce the traffic.Another comment is expected to relate the present work to the separated administrations, techniques for making diverse level of administration for client willing to pay elevated amounts. As more items are made to control system, separated administration will turn out to be critical for the internet. Recognisable proof of source that can be compelled to retrain their flow rate can prompt present distinctive need of activity and responsible flow. For instance, if activity from a specific machine is regarded high need, the framework may confine different machine, rather than backing of high need machine.

The description for a diversity of traffic from each mechanism, each with a dissimilar precedence changing the architecture to work on a flow basis as an alternative of a mechanism basis could also be straightforwardly done. As a result the techniques that we apply in this paper are honestly appropriate to the best effort traffic in excess of the Internet, but their extensions to differentiated Services or QoS environments are directly ahead.

2. RELATED WORK

Internet users are rapidly growing day by day, resulting in network management becoming one of the important factors in today communication network. One of the challenges deals with how network managers are accomplishing prevention of networking problems instead of repairing them. When the network problems occur many enterprises, individuals and organizations face terrible breaks which are dependent on network connection. Hence they present a viable method of implementing a learning mechanism that uses neural network for data communication network. To achieve above average result they used actual data network for creating a connection between layers of neural network and chose only important nodes for designing a neural network [1].

Many researchers have used random early detection algorithm for better quality of service and a new way of congestion control because quality of service is very important factor in real time or time sensitive applications like voice and video. TCP / IP differentiated service fulfil the above target by using neural network and implement a new TCP /IP variant i.e. NRED [2]. NRED is well suited technique for homogenous and heterogeneous allocation service without any special modification.

Heterogeneous receivers always suffer the problem of large propagation delay which would a harmful factor for efficient congestion control scheme in data transfer. Feedback arrival at the source in right time and right way is useful for congestion control action that applied at the time of congestion but if the feedback is outdated then mismatch occurs between network resources and amount of admitted traffic. To solve above mismatch problem novel congestion control technique is described that is based on back propagation neural network [5].

Wireless mesh network experiences lossy nature of the communication network which is challenging problem for reliable transport protocols. Conventional TCP variants do not deal with this lossy nature during congestion control in communication network. To avoid this problem a novel neural network based congestion control technique proposed.

And for detailed analysis of this proposed technique implemented which is i-TCP or intelligent TCP [3]. Their results showed a significant improvement in total network throughput and average energy consumption per bit as compared to other congestion control techniques used in other variants of TCP.

TCP experiences a non-linear and time varying property that makes network dynamic resulting queue management for congestion avoidance techniques are incapable of sufficiently adapting TCP network dynamics. Non-linear neural network controller using non-linear model of TCP were proposed [4]. They were using genetic algorithm to train the non-linear neural controller. This technique generated better result as compared to other techniques for congestion control with faster transient response, bigger throughput and higher link utilization.

In Ad hoc wireless network there is no wired or reliable connection for communication resulting ad hoc network experiences a lossy medium for data transmission. In ad hoc network there are several packet losses due to link bit error but not due to congestion. TCP should be unable to differentiate the reason of packet loss and considers congestion and apply congestion control method. The right reaction of TCP during link bit error should be resending the loss packets instantly but not reducing the data sending rate. Since avoiding this above trouble a back propagation neural network classifier is used for distinguish network congestion and link bit error [6].

Our main contribution is the development of a neural network predictive technique for the congestion detection to maintain the quality of service. Through simulation of trace data from practical status of data loss we demonstrate the technique improve detection performance. This paper not only introduces a new model and a neural network, but also analysis the neural network architecture and evaluates detection performance.

3. CONGESTION CONTROL MODEL

The model used is the congestion detection model which detects congestion in the network. Network consists of various nodes and this contains different types of packets statistics. Based on the different types of packet, its characteristics and some added parameters to evaluate congestion. We use different components in our model, this architecture of the model and its various components are explained as follows:

3.1 Architecture

A data communication network generally consists of a number of source and destination that are geographically distributed. Packets generated at a source node are delivered to their target through a series of intermediary nodes. The paper considers an architecture revealing a network which has control agents that are placed somewhere on a node in that network. This control agent has together the power to read from and to supremacy network nodes. The nodes concerned would either account the essential statics to the control agent or the control agent would poll these nodes.

3.1.1 Data Network

To develop a communication network, consider many network nodes in a configuration where all of the considered nodes were attempting to send data to one other node with the help of intermediary nodes. Each node attempts to send at random bit rate. A random amount of variance is given to each node's allocation rate for better illustration. Link capacities among all nodes in network were given subjective values for trying reason.

3.1.2 Control Agents

In our communication network we generated a control agent which contains a neural network. In our simulation the control agent is, on the go at a usual time period in a piece of simulation code. This enables the agent to examine without difficulty and influence traffic data from each and every node. The control agents collect information from each managed nodes, do a number of mathematical function normalizing the values, and create a conclusion about where network trouble will occur.

3.2 Implementation Detail

The scheme consists of three separate programs, one implemented in Tcl, the language used to run simulation in NSG, one implemented in Matlab, for feature and data extraction and the third essentially running in the neural network. We use the widely available Matlab application as our neural network.

3.3 The Simulation Network

In our configuration node we created network where all sending nodes are connected to each one of the receiving nodes through many links which directly send packets to the destination. In our simulation network we used 50, 100, 150, 200 nodes which produces data in a technique like to UDP agents transferring CBR traffic with a randomized parameter to insert discrepancy to the traffic.

In our simulation network every node are linked to their particular receiving node. Packet interval and packet size are specified in the simulation script which decide how rapid the sender sends data, which provides information the simulation run or execute. The sender sends a packet of the elected amount at the elected time. The receiver sends no reply which revenue receiver just has a null agent that receives the data. Initial we produced a simulation network which had 50, 100, 150 and 200 nodes and their simulation time and packet speed are 10, 20, 30, 40 and 1.0 mbps and 5.0 mbps respectively. After simulating all nodes, trace file and Tcl file were generated and applied pre processing data transformation. We gather statistics such as packet received, packet dropped, packet forwarded and packet send during the simulation. The control agents observe traffic and production assessment. We used a single hidden layer, feed-forward neural network and neural network prediction element for our agent accomplishment.

3.3.1 Matlab Specifics

We find trace file for all simulated network with the help of Tcl file and extracted data for each and every node such as DROP, SEND, RECV, FORW data. This above extracted data is our significant network congestion related criteria or terms to develop our congestion detection system. These above extracted criteria are stored in mat file written by simulator and contain current values for the number of packets. We also used feature extraction for all these DROP, SEND FORW and RECV criteria such as total drop per packet, average per packet drop, cumulative per packet drop, minimum packet drop and maximum packet drop for all above criteria's. These values are then normalized for the neural network.

3.3.2 Neural Network Specifics

In our project the neural network was used, 5*n input nodes, one hidden layer containing n nodes and n nodes in the output layer. By using feature extraction process 'total' file was generated and this file used as input file which have total drop per packet, average per packet drop, cumulative per packet drop, minimum packet drop and maximum packet drop and similar to SEND, FORW, and RECV operations. We generated clas file which have two terms first for few data loss and second for extreme data loss and this clas file was used as an output file. In our project connectivity is based on adjacency for represented node from input neurons up to the hidden layer, in its place of providing a completely associated atmosphere between the input and the first hidden layer.

3.4 Working Process of Model

The model proposed in this paper**,** is used to detect congestion in the network. The network may consist of various numbers of nodes. The nodes may have different types of packets statistics like packet drop, packet sent, packet forwarded and packet received. We will evaluate each node in the network where congestion is to be detected on the basis of these types of packets statistics.

We will evaluate packet statistics on the basis of terms like min, max, cumulative per packet, average per packet, and total packets. As a whole the process is called feature extraction. After "feature extraction" comes the "feature collection" process in which data is collected. As a result of feature extraction here we had 27 cases (here is the network in the form of files to be tested) of network simulations which after simulation results in the number of congestions and number of non-congestion and the result will be generated.

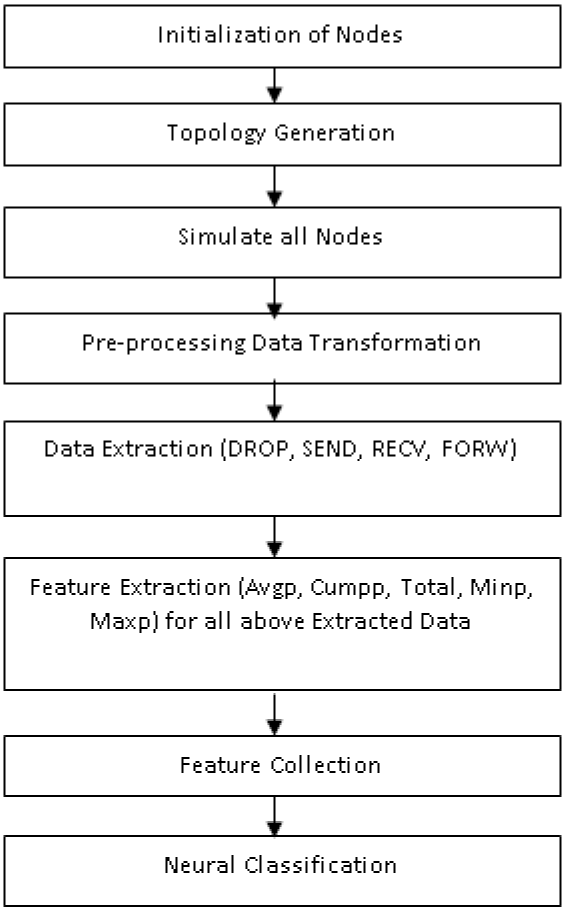

Figure 1 Working Process of Model

4. SIMULATION ENVIRONMENT

4.1 Test Cases

The tests were performed on a matlab R2013a running window 10. The packet size was set to 1500 bytes for every transferring node. All associations between nodes were set to 1.0 MB\s and 5.0 MB\s. The traffic generated is constant bit rate traffic, but the arbitrary constraint was set for every transferring node so that the traffic would not be completely stable.

5. SIMULATION RESULTS

A general result can be found in the below graph which shows our current congestion detection system detects congestion and non congestion levels in various locations in the network in about 99% of the cases.

We used 10 fold validation processes for network simulation. Roughly 40.7% of the cases were simulation of a network without congestion problem. About 59.3% of the total cases had various level of congestion in various locations in the network.

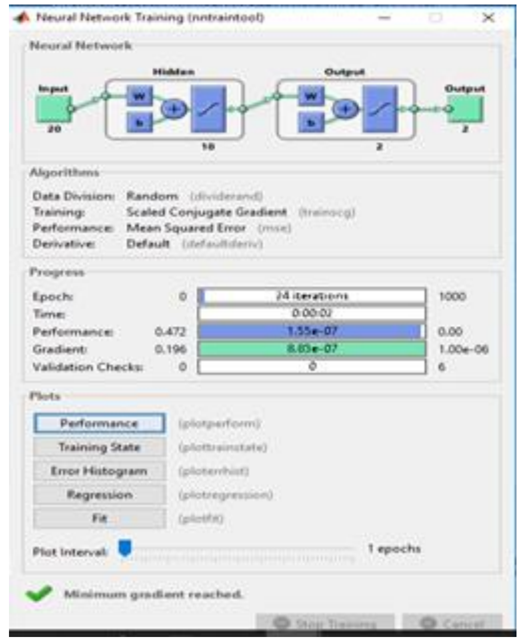

Figure 2 Neural Network Training and Testing

Figure 2 shows training of neural network window with 10 hidden layers for training. The total number of epochs used is 1000 with 24 iterations. Further statistics illustrate performance, time, validation checks and gradient used for testing and training of neural network.

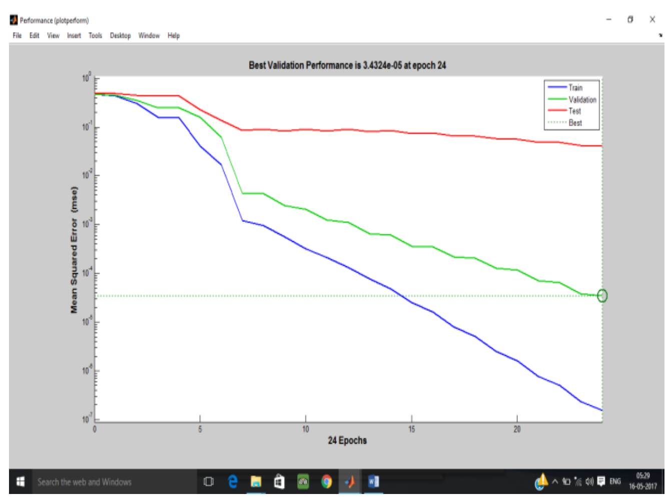

Figure 3 Validation performances after 24 epochs.

Figure 3 shows validation performance through admiration to epochs. It obvious from figure that as we remain on growing the number of epochs for training and testing the error take keeps on falling. After 24 epochs the assessment is 3.4324e-05.

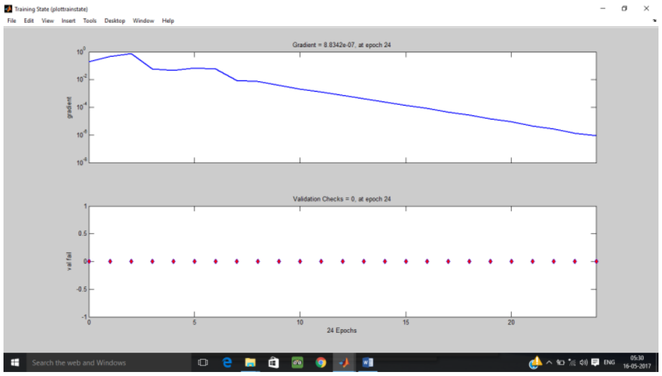

Figure 4 Training state or plot train state diagram.

Figure 4 shows the validation in gradient coefficient with respect to number of epochs. The concluding value of gradient coefficient at epoch number 24 is 8.8342e-07 which is around near to 9. Least the assessment of gradient coefficient improved will be training and testing of the network.

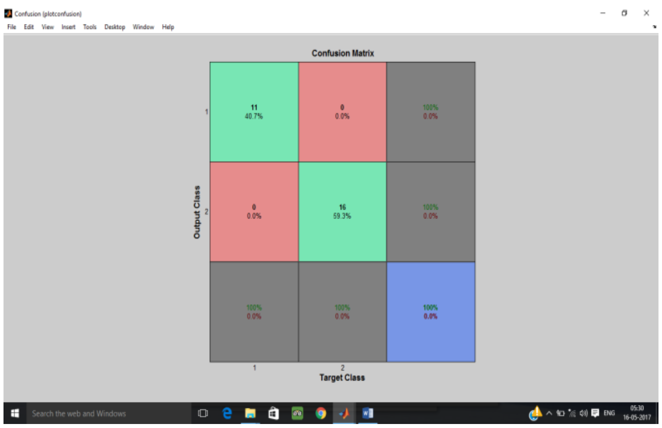

Figure 5 Confusion Matrix

The above confusion matrix shows the 99% accuracy in our dataset. It shows true negative is 40.7% and true positive is 59.3% accuracy of our system.

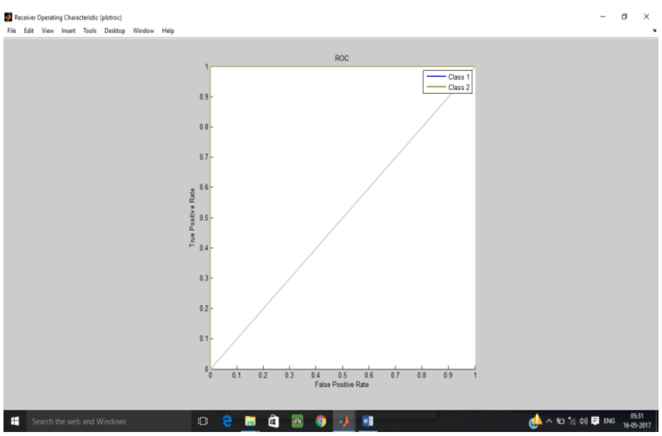

Figure 6 Receiver operating characteristic (ROC) graph

Figure 6 shows the Plot ROC function and plot ROC function calculates the true positive rate and false positive rate. Above graph shows great accuracy in our system.

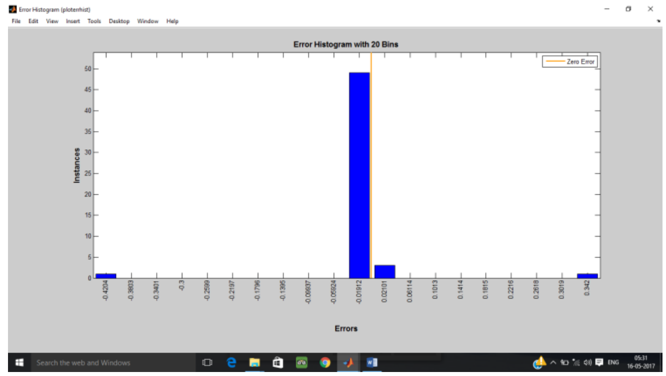

Figure 7 Error histogram with 20 bins

The above Figure 7 shows the zero error which means our system provides 99% accuracy.

6. CONCLUSION

This paper has described a congestion detection mechanism for data communication network using neural network predictive technique. We have explored through the use of network simulator, that a neural network can achieve great accuracy in predicting one particular network problem, namely congestion. Our approach is very beneficial for many more network problems but predicting congestion is just the first step towards our research goal. The simulation results demonstrate that the proposed neural network predictive technique works well from the point of the system response and accuracy.

ACKNOWLEDGEMENT

I want to thank my guides Prof. Sumit Sar and Prof. Monika Arya and the department of computer science & Engineering of Bhilai Institute of Technology, Durg, India for their motivation, encouragement and support towards the completion of this project.

REFERENCES

[1] J. Alan Bivens, Boleslaw K. Szymanski, Mark J. Embrechts,“ Network congestion arbitration and source problem prediction using neural network”, Smart Engineering System Design, Vol. 4:243-252, 2002.

[2] Tran Xuan Truong, Nguyen Duy Viet, Le Hung Lan and Mai Vinh Du,“ Congestion Control In TCP/IP Differentiated Services Network Using neural network”, SeventhIEEE International Conference on Natural Computation, 2011.

[3] A. B. M. Alim Al Islam and Vijay Raghunathan, “ End-to-End Congestion Control in Wireless Mesh Network Using a Neural Network”, IEEE WCNS, 2011.

[4] Modjtaba Rouhani, Mohammad RasoulTanhatalab, Ali Shokohi-Rostami, “Non-linear Neural Network Congestion Based Genetic Algorithm for TCP/IP Network”, Second IEEE International Conference on Computational Intelligence, Communication System and networks, 2010.

[5] N. Xiong, Y. Yang, Jing He and Yanxiang He, " On Designing QoS for Congestion Control Service Using neural Network Predictive Technique", IEEE, 2006.
[6] Gong Changqing, Zhao Linna, Wang Xiaoyan, "Using Neural Network Classifier of Packet Loss Cause to Improve TCP Congestion Control Over Adhoc Networks", IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communication, 2007.
[7] George A. Rovithakis and Christos N. Houmkozlis, " A Neural Network Congestion Control Algorithm for the Internet", IEEE International Symposium On Intelligent Control Limassol, June 2005.
[8] YamineeSahu, Sumit Sar, "Congestion Control Analysis in Network: A Literature Survey", International Research Journal of Engineering and Technology (IRJET), Vol. 3 Issue 11, 2016.
[9] Hyun C. Cho, M. Sami Fadali, Hyunjeong Lee, " Neural Network Control for TCP Network Congestion", IEEE American Control Conference, Portland, USA, June 2005.
[10] ShilpiKaura and A.K. Vatsa, "A Novel Architecture and Mechanism for Congestion Control in High Speed Network", International Journal of Next Generation Networks (IJNGN), Vol. 3, No. 1, March 2011.
[11] Sachin Kumar Saxena, Dhaneshwar Kumar, Garima Krishna, Neha Goel, Sati Ayush, " Congestion Control in Wired Network for Heterogeneous Resources Using Neural Network", International Journal of Advance Research in Computer Science and Software Engineering, Volume 3, Issue 5, May 2013.
[12] Parisa Bazmi and Manijeh Keshtgary, " A Neural Network Based Congestion Control Algorithm for Content Centric Networks", Journal of Advance Computer Science and Technology, 3 (2) 2014.
[13] Dr. T. Karthikeyan and B. Subramani, " A New Congestion Control Algorithm Based on Noval AQM", International Journal of Computer Science and Engineering (IJCSIT), Vol. 5(3), 2014.

Authors

Yaminee Sahu is research scholar at the department of computer science and engineering of Bhilai institute of technology, Durg. She has authored one paper in international journal and has been actively serving as a reviewer for international journals and conferences. Her current research areas include ad-hoc and sensor network, wireless sensor network and computer networks.

Sumit Kumar Sar is an associate professor at the department of computer science and engineering of Bhilai Institute of Technology, Durg. He has authored several papers in international journal and has been actively serving as a reviewer for international journals and conferences. His current research areas include neural network, Fuzzy logic and Image processing.