A Scheme of Anomalous Detection Based on Reinforcement Learning for Load Balancing

Hye-Young Kim*
Major in Game Software, School of Games, Hongik University, S. Korea
*Corresponding author

Abstract. In recent, both researchers and developers have great interests in anomalous detection. However, it is still difficult to implement a uniform framework for anomalous detection. Also, the network anomalous detection using deep learning methods has been discussed with potential limitations and interests. An anomalous detection in wireless or wired network is extremely important because it is caused by flood traffic of network and intrusion. Patterns of malicious network loads are defined, while anomalous detections is more suitable for detecting normal and anomalous network loads by means of deep learning. The important goal of these issues is to recognize the anomalous detections for better preparation against future load balancing of networks. In this paper, we propose an agent Detectbot that processes anomalous detection for load balancing based on reinforcement learning. Our simulation results show that the reinforcement learning scheme is effective for anomalous detection in load balancing.

1. Introduction
The definition of a deep learning has a variety of definitions such as unsupervised machine learning, artificial intelligence, learning multiple layers [1]. A great deal of attention has been given to deep learning over the past several years for many applications in diverse areas, such as image processing, natural language processing, computer vision, and so forth [2, 3]. The loads of network are deployed to monitor one or more events in an unattended environment. A large number of the event data will be generated over a period of time in network environments. So, a scheme of anomalous network detection for load balancing based on deep-learning is critical issues in network environments. There is great interest in anomalous detection in researches [4-7].

The deep learning area is still in an infantile stage, with most researchers still experimenting on combining various algorithms and layering approaches to produce the most accurate and efficient solution for a specific dataset [8]. In this paper we focus on investigating deep learning algorithms employed for anomalous network detection. And, we propose an agent Detectbot that processes anomalous detection for load balancing based on deep learning. Also, the Detectbot measures network load and processes structural configuration by analyzing a large amount of user data and network load, and applies Deep Learning’s Deep Belief Network method in order to achieve efficient load balancing in network environments. We address the key functions for our proposed scheme and simulate the efficiency of our proposed scheme. Our proposed deep learning model can make a valid contribution to both deep learning and reinforcement learning. This is based on the reinforcement learning with experience replay for achieving the maximum prediction accuracy continuously in online learning systems while detecting network intrusions.
The rest of the paper is structured as follows. Section 2 includes the background introduction about open load balancing and reinforcement learning issues. In Section 3, we describe our proposed architecture and provide the detailed anomalous network detection scheme which proposes the agent Detectbot that processes anomalous detection for load balancing based on reinforcement learning, a measure of network load, structural configuration, a neural load prediction algorithm, and balance of the load, in order to achieve efficient load balancing in network environments. In the final section, we constitute a summary of our proposal.

2. Related Works

In [9], various methods for malicious code detection are introduced such as naive bayes, decision trees, Support Vector Machines (SVM), and so on through literature research. However, the major problem was that feature extraction was not appropriately applied to malicious detection so that it caused low detection rate and accuracy. For this reason, this research proposed a hybrid malicious code detection model in conjunction with Auto-Encoder and Deep Belief Networks. In this research, Auto Encoder that consists of three steps such as pre-training, unrolling, and fine-tuning was used for dimensionality reduction, and Restricted Boltzmann Machine was also used in this Auto Encoder methodology as a pre-training step. One important concept was that the RBM algorithm was driven by the energy formula and the weight updating formula as described earlier [10].

Denial-of-Service (DoS) attack floods a host website with a huge number of requests to make it slow in Remote-to-Local (R2L) attack a remote where a remote hacker tries to get local user privileges in User-to-Root (U2R) attack a hacker operates as a normal user and exploits vulnerabilities in the system [11] in the Probing attack a hacker scans the machine to determine vulnerabilities which could be exploited later. Totally there are 148, 516 connections, of which 53.5% are normal and the rest are malicious. The dataset contains 41 features, of which 32 are continuous, 6 are binary and 3 are categorical. For the purpose of this study, we converted like in the previous research [12] all the continuous features into categorical using binning, and then converted all the categorical variables to a binary representation using one hot encoding.

Salama et al. [13] conducted research for the anomaly intrusion detection scheme using RBM-based Deep Belief Network (DBN). In this research, SVM classifier was followed by DBN which was used for feature reduction so that a hybrid scheme of DBN and SVM was the primary research methodology. This methodology consisted of three main phases: pre-processing, DBN feature reduction, and classification.

Anomaly detection has been applied in countless application domains such as medical and public health, fraud detection, sensor networks. Anomaly detection is an important data analysis task that detects anomalous or abnormal data from a given dataset. It is an interesting area of data mining research as it involves discovering and rare patterns in data. It has been studied in statistics, machine learning, detection, and exception mining. An anomaly is an observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism. Anomalies are considered important because they indicate significant but rare events and can prompt critical actions to be taken in a wide range of application domains [14].

A network anomalous detection has been more challenging than it was before. As most existing supervised techniques are based on knowledge provided by an external agent, they require labelled data and are unable to detect vulnerabilities. Some researchers focused on the detection principle and operational aspects. In [15], the taxonomy matrix of intrusion systems is proposed and the matrix is formed from the output type and data scale elements of the taxonomy. Another survey considered anomaly detection methods for network intrusion detection in wired networks [16]. Most recently, it classified outlier detection techniques as supervised and unsupervised but their taxonomy is slightly confusing because they consider proximity based approaches under the supervised category [17]. Nowadays, deep learning has been widely studied, since it learns features automatically from raw data. In many computer vision applications, deep learning has shown impressive performance, such as image segmentation, object detection, and activity recognition. These works mainly focus on supervised scenarios. However, in the field of anomaly detection, labelled abnormal events are seldom available for training. Fortunately, unsupervised deep learning approaches have also been studied in
recent years to address important tasks, such as image classification and object tracking. The reason why deep learning conducts inspiring performance is that multi-layer non-linear transformations can adaptively extract meaningful and discriminative features. Nevertheless, deep learning is seldom studied for abnormal event detection.[18]

Network anomaly detection methods are generally tested using datasets developed at the end of last century, justified by the need for publicly available test data and the lack of any other alternative datasets. Widely accepted as benchmark, these datasets no longer represent relevant architecture or contemporary attack protocols, and are accused of data corruptions and inconsistencies. Hence, testing of network anomaly detection techniques using these datasets does not provide an effective performance metric, and contributes to erroneous efficacy claims [14].

By exploring potential rules among normal trajectories, abnormal events are identified as ones which disobey these rules. In [19], the authors extract multiple features based on trajectories, such as the trajectory mean, speed and acceleration. In their method, each feature is applied with a clustering algorithm, and the final clustering result is obtained by taking clusters from all features into account. The clusters with few members and samples far away from these cluster centers are treated as anomalies. The benefits arising from the services controlled by communication between objects are now being increased by people who use these services in real life [19]. Hence, a load balancing protocol and anomalous detection of the load of networks are critical to considerations in the design of the networks. Therefore, we propose an agent, Detectbot that measures network load and processes anomalous detection for load balancing based on deep learning by analysing a large amount of user data and network load. Moreover, in order to achieve efficient load balancing, we propose applying Deep Learning’s reinforcement learning method. Additionally, we address the key functions for our proposed scheme and simulate the efficiency of our proposed scheme using mathematical analysis.

3. Proposed Scheme

In order to model complicate nonlinear relationship of network, we have drown out the structure map of network load by using reinforcement learning which is graph generating model connected to each layer by in-depth connection between multiple layers which created nerve network.

![Figure 1. Deep reinforcement learning.](image)

In Figure 1, we show the implementation of the deep reinforcement learning. Many of the successes in deep reinforcement learning have been based on scaling up prior work in reinforcement learning to high dimensional problems. This is due to the learning of low dimensional feature representations and the powerful function approximation properties of neural networks. By means of representation learning, deep reinforcement learning can deal efficiently with the curse of dimensionality, unlike tabular and traditional non-parametric methods [20]. In general, deep reinforcement learning is based on training deep neural networks to approximate the optimal policy $\pi^*$, and/or the optimal value functions $V^*$, $Q^*$ and $A^*$. The well known function approximation properties of neural networks led naturally to the use of deep learning to regress functions for use in reinforcement learning agents. There is an input vector $x$
∈ R^d, which maps it to the latent representations h_i ∈ R^a_i, where d represents the dimension of the vector using a deterministic function in equation (1). There is an input vector x ∈ R^d, which maps it to the latent representations h_i ∈ R^d, where d represents the dimension of the vector using a deterministic function in equation (1).

\[ h_i = \sigma(W_i \cdot h_{i-1} + b_i); \quad i = 1, \ldots, n, \]  

(1)

where h0 = x, σ is an activation function (the sigmoid function \( \sigma(t) = 1/(1 + e^{-t}) \)) and n is the number of hidden layers. And its output vector is calculated as the latent representation.

\[ y = \sigma(W_{n+1} \cdot h_n + b_{n+1}) \]  

(2)

Where the estimator of the model \( \theta = (W, b) \) by minimizing the square reconstruction error over m training samples \((x(i), y(i))_{i=1}^{m}\), as shown in equation (3)

\[ E(\theta) = \sum_{i=1}^{m} m (x(i) - y(i))^2. \]  

(3)

---

**Figure 2. Proposed scheme.**

In Figure 2, our proposed scheme is trained using the back propagation algorithm as follows.

1. Initialize \( Q(s, \alpha) \) and start from a random state from a time-based sequence dataset.
2. Forward the pass of stats and get Q-values for 2 actions, anomaly and not anomaly.
3. Perform a \( \varepsilon \)-greedy exploration for choosing an action \( \alpha \) for the current states, either a random action with a small probability \( \varepsilon \), or the largest maximum Q-value for states.
4. Pass forward the next states’ with the action \( \alpha \) and calculate the reward \( r \), \( E(\theta) = \sum_{i=1}^{m} m (x(i) - y(i))^2. \)
5. Store the transitions from the current to the next state with the reward \( r \) and the action \( \alpha \).
6. Update the weights performing the gradient descent \((r_j + \gamma \max_{\alpha} Q(\phi_{j+1}, \alpha)) - Q(\phi_j, \alpha; \theta))^2 \]
7. Repeat the steps 2-6 for being trained.

Once enough episodes of exploration are completed, our proposed scheme would have adjusted its weights to predict correct \( Q(s, \alpha) \) values, so that the agent would be able to act with optimal policies in any given state. Our proposed scheme to learn \( Q(s, \alpha) \) values is implemented using TensorFlow[21] and Scikit Learn[22].

Our purpose scheme is to teach an autonomous agent with minimal human interference to flag malicious network connection requests as anomalous. This is done by providing a positive reward when the agent makes correct actions, such as flagging malicious network connection requests and not flagging normal connection requests, and a negative reward when the agent makes wrong actions such as either flagging normal connection requests or not flagging malicious connection requests. Likewise
the most existing deep learning research [17], our proposed DQN was implemented using TensorFlow and Scikit Learn for the 10-fold cross-validation libraries.

4. Conclusion
In this paper, we propose the agent Detectbot that processes anomalous detection for load balancing based on deep learning. Also, the Detectbot measures network load and processes structural configuration by analysing a large amount of user data and network load, and applies reinforcement learning method in order to achieve efficient detection of anomalous network. We address the key functions for our proposed scheme and simulate the efficiency of our proposed scheme. In the future researches, we will show the effectiveness of ours, and compare with previous researches.

Acknowledgments
This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2019R1A2C1008533), Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. 2016R1A2B4012386) and this work was supported by 2019 Hongik University Research Fund.

References
[1] Deng, L., Yu, D., "Deep learning methods and applications. Found," Trends Signal Process, 7(3–4), 199–201, 217 (2014).
[2] Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M., Seliya, N., Wald, R., Muharemagic, E., “Deep learning applications and challenges in big data analytics”, J. Big Data 2(1), 1 (2015).
[3] Deng, L., Yu, D., ”Deep learning: methods and applications,” Trends Signal Process. 7(3–4), 197–387 (2014).
[4] Peter O’Connor, Daniel Neil, Shih-Chil Liu, Tobi Delbruck and Michael Pfeiffer, “Real-time classification and sensor fusion with a spiking,” Frontiers in Neuroscience, vol. 17, article 178, pp. 61-73, October 8, 2013.
[5] Mostafa A. Salama, Aboul Ella Hassanien, Aly A. Fahmy, “Deep Belief Network for clustering and classification of a continuous data,” The 10th IEEE International Symposium on Signal Processing and Information Technology, Dec. 2010, p.472-477.
[6] Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton, “Imagenet classification with deep convolutional neural networks,” In Advances in Neural Information Processing Systems 25, pp.1106–1114, 2012.
[7] Hamid Maei, Csaba Szepesv’ari, Shalabh Bhatnagar, and Richard S. Sutton, “Toward off-policy learning control with function approximation,” In Proceedings of the 27th International Conference on Machine Learning (ICML 2010), pp. 719–726, 2010.
[8] D. Xu, E. Ricci, Y. Yan, J. Song, N. Sebe, “Learning deep representations of appearance and motion for anomalous event detection,” Proceedings of the British Machine Vision Conference, 2015, pp. 1–12.
[9] Li, Y., Ma, R., Jiao, R.,” Hybrid malicious code detection method based on deep learning,” Int. J. Secur. Appl. 9(5), 205–216 (2014)
[10] Donghwoon K., Hyunjoo K., Jinoh K., Sang C. S., Ikkyun K., Kuinam J. K.,”A survey pf deep learning-based network anomaly detection,” Cluster Computing, 27 Sep. 2017
[11] Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.,”A detailed analysis of the KDD CUP 99 dataset,” Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), 53–58 (2009)
[12] NSL-KDD dataset, Canadian Institute for Cyber Security, University
[13] Salama, M.A., Eid, H.F., Ramadan, R.A., Darwish, A., Hassanien, A.E., “Hybrid intelligent intrusion detection scheme,” Soft Comput. Ind. Appl. 96, 293–303 (2011)
[14] Mohiuddin Ahmed, Abdun Naser Mahmoud, Jiankun Hu, “A survey of network anomaly detection techniques,” Journal of network and computer applications, 60(2016), pp.19-31
[15] Furnell S, Tucker C, Furnell S, Ghita B, Brooke P., “A new taxonomy for comparing intrusion detection systems,” Internet Res. 2007, 17(1), pp. 88–98
[16] Estevez-Tapiador JM, Garcia-Teodoro P, Diaz-Verdejo JE., “Anomaly detection methods in wired networks: a survey and taxonomy,” Comput Commun 2004, 27(16), pp.1569–84
[17] Gogoi P, Bhattacharyya D, Borah B, Kalita JK.,"A survey of outlier detection methods in network anomaly identification,” Comput J, 2011, 54(4), pp. 570–88
[18] Y. Feng, Y, Yuan and X. Lu, “Learning deep event models for crowd anomaly detection,” Neurocomputing, 219 (2017), pp.548-556
[19] N. Anjum, A. Cavallaro,” Multifeature object trajectory clustering for video analysis,” IEEE Trans. Circuits Syst. Video Technol.,18 (11) (2008), pp.1555–1564
[20] N. Wang, D.-Y. Yeung, “Learning a deep compact image representation for visual tracking,” Proceedings of the Advances in Neural Information Processing Systems, 2013, pp. 809–817
[21] Tensorflow, http://github.com/tensorflow/tensorflow
[22] Scikit Learn, http://scikit-learn.org/stable