aurum

AURUM MÜHENDİSLİK SİSTEMLERİ VE MİMARLIK DERGİSİ
AURUM JOURNAL OF ENGINEERING SYSTEMS AND ARCHITECTURE

Cilt 2, Sayı 1 | Bahar 2018
Volume 2, No. 1 | Spring 2018, 115-123

NETWORK INTRUSION DETECTION USING MACHINE LEARNING TECHNIQUES

Oğuz ATA[1]
[1] Altınbaş University, Software Engineering, Istanbul
Oguz.ata@altinbas.edu.tr

Khalid KADHİM[2]
[2] Altınbaş University, Electrical and Electronics Engineering, İstanbul
altaeekh@yahoo.com

Abstract

Recently, it has become important to use advanced intrusion detection techniques to protect networks from the developing network attacks, which are becoming more complex and difficult to detect. For this reason, machine learning techniques have been employed in the Intrusion Detection Systems (IDS), so that, more complex features can be detected in the characteristics of the packets incoming to the network. As these techniques require training data, many datasets are collected for this purpose. Some of these datasets have known issues that limit the ability to apply intrusion detection systems built, based on these datasets, in real-life applications.

In this study, the existing intrusion datasets are illustrated alongside with the known issues of each dataset, as well as, the existing intrusion detection systems that employ machine learning techniques and use these datasets, are discussed. As machine learning techniques extract different knowledge from different datasets, and each technique has different approaches to extract that knowledge, the performance of each technique is different from one dataset to another. The results of the discussed studies show the great potential of using machine learning techniques to implement IDS, where the Artificial Neural Networks (ANN) have shown the highest average performance, among other machine learning techniques.

Keywords: *Machine Learning, Artificial Neural Network, Intrusion Detection Systems.*

MAKİNE ÖĞRENMESİ TEKNİKLERİ KULLANILARAK AĞ SALDIRI TESPİT SİSTEMİ

Özet

Son zamanlarda gelişen ağ saldırılarından korunmak için saldırı tespit sistemler önemli bir hale gelmiştir. Bu saldırılar, öncekilerden daha karmaşık ve tespit edilmesi zordur. Bu nedenle Makine Öğrenmesi teknikleri kullanılmaya başlanmıştır. Böylece ağdan gelen paketlerin karekteristiklerinde, daha karmaşık özellikler tespit edilebilmektedir. Bu teknikler öğrenebilmek için belirli özelliklerde verysetine ihtiyaç duymaktadır. Bu amaç ile birçok very seti toplanmıştır. Bu verisetlerinin bazıları gerçek hayat uygulamalarında saldırı tespit sistemlerinin uygulamasında bilinen limitlere sahiptir.

Bu çalışmada Bu her bir veri setinin bilinen konularının yanı sıra, makine öğrenim tekniklerini kullanan ve bu veri setlerini kullanan mevcut saldırı tespit sistemleri ile birlikte herbir mevcut izinsiz veri kümeleri de tartışılmıştır. Makine öğrenme teknikleri farklı veri kümelerinden farklı bilgi çıkarımında bulunurlar ve her tekniğin bu bilgiyi elde etmek için farklı yaklaşımları olduğu için, her tekniğin performansı, bir veri kümesinden diğerine farklıdır. Tartışılan

çalışmaların sonuçları, Yapay Sinir Ağları (YSA) 'nın diğer makine öğrenme teknikleri arasında en yüksek ortalama performansı gösterdiği görülmüştür. Böylece Saldırı tespit sistemi uygulamaları için makine öğrenme tekniklerini kullanmanın büyük potansiyeli olduğu görülmüştür.

Anahtar Kelimeler: *Makine Öğrenmesi, Yapay Sinir Ağları, Saldırı Tespir Sistemi.*

1. INTRODUCTION

Concerns about the security of networks are rising in the recent years, according to the rapid development of the techniques used to attack these networks. According to his development, detecting traffic incoming from intrusion attempts is becoming more difficult, as the techniques used in these attacks attempt to use network packets similar, in characteristics, to those incoming from normal traffic, which makes traditional networks protection techniques very weak toward such attacks. Thus, more complex techniques are being developed to protect these networks against complex attacks, such are the use of machine learning to distinguish packets of normal traffic from those from attacks [1, 2].

Machine learning is the field of study that aims to provide computers with the ability of gaining knowledge from the external world, without any human interaction. The knowledge extracted by a certain machine learning technique may be different from one set of inputs, from the external world, to another. Moreover, knowledge extracted from a single set of inputs may also be different from one machine learning technique to another, according to the different approaches used to extract such knowledge. One of the main machine learning fields is data mining, where the inputs from the external world are datasets, collected from the domain that knowledge extraction is required for [3].

Data mining techniques, as well as other machine learning techniques, are categorized into two main categories, which are unsupervised and supervised techniques. Unsupervised data mining techniques require no addition to the input dataset, as the aim of these techniques is to extract relations among the objects in the dataset, while supervised data mining techniques require some extra information to be added to the dataset by an expert. Supervised data mining techniques extract the relations among the objects in the dataset, and the knowledge added by the expert. This knowledge, extracted from the sample dataset, which is known as the training dataset, can be used in runtime to apply the extracted knowledge on new objects to assist the operation of the system interacting with the domain [4].

Classification is one of the widely used data mining techniques, where the information added to the dataset in the form of labels that classify each object in the dataset into one, or more, of the classes that exist in the domain. During the training phase of the classifier, the characteristics of objects in each class are extracted, so that, a model is built by the classifier base on these relations. These models are then used to predict classes for new data objects during the runtime in order to assist the decision making for the system controlling the domain. These decisions are based on the characteristics of the category that the data object is predicted to be in. Thus, different Intrusion Detection Systems (IDS) are proposed based on classification techniques, where data are collected from network traffic that includes normal and attack packets in order to train classifiers to be able to predict a class for each object to allow denying attack packets from accessing the network [5].

2. NETWORK TRAFFIC DATASETS

Different datasets are collected for the characteristics of packets in network traffic that includes normal and attack packets to build and evaluate the performance of data mining techniques in the field of network protection. As the classifiers have different approaches to extract knowledge from datasets, it is important to evaluate the performance of the classifier, as a measure of the quality of the extracted knowledge. However, as the classifiers are used to provide predictions, labeled data are used for that evaluation, by comparing the predictions provided by the classifier to the actual classes, or labels, that these objects belong to. Thus, each dataset is split into two parts, one is used by the classifier to extract the knowledge, which is known as the training dataset, while the other is used to evaluate the performance of the classifier, by comparing the predicted classes to the actual ones, which is known as the testing dataset [6].

One of the earlier dataset collected for network traffic to train and evaluate data mining techniques in IDS is the KDD Cup'99 dataset [7]. This dataset includes information about 4,898,431 network packet, where each packet is characterized using 41 different features. Each packet is labeled with one of five labels, one for normal packets, and four for different network attacks that are included in the dataset, which are:

1. Probing Attack: is an attack that aims to gather any possible information about the network and the computers that belong to that network in order to use that information to compromise the security of the network.

2. User to Root Attack (U2R): is an attack that exploits information of users who have legitimate access to that network to gain root access to the system using any vulnerability in that system

3. Denial of Service Attack (DoS): is an attack that attempts to exhaust the resources available on a computer, such as memory or processing power, in order to deny providing services to legitimate users.

4. Remote to Local Attack (R2L): is an attack where the attacker has access to the network but does not have the necessary information to authenticate to the services provided on that network.

Although this dataset is widely used to train and evaluate many data mining-based intrusion detection systems, different issues that this dataset suffers from are illustrated by Tavallaee et al. [8] and McHugh [9]. The first issue stated in this dataset is the tools used to collect the packets' information, such as the TCPdump tool which is expected to drop some of the network traffic during heavy traffic. Such drops are not examined during the collection of the dataset. Another issue is the definition of attacks included in the dataset, where probing attacks, for example, are not considered actual attacks, unless a certain threshold is exceeded, where such conditions are not considered in the data collection. Moreover, the number of redundant objects in the dataset is extremely high, which affects the difficulty of the analysis, as objects similar to those in the testing dataset are highly expected to be in the training dataset, which reduces the difficulty of predicting classes for these objects.

A newer version of the KDD CUP'99 dataset is proposed, which is known as the NSL-KDD dataset with the same classes of the earlier version are used in this version of the network traffic dataset. Although the redundancy issue of the data objects in the dataset is fixed in this version, concerns about the application

of synthetically collected data in real-life domains are raised, as well as the setup of the environment setup that is used to collect this data, which is described to be questionable by Shakil and Dewan [10].

Nour Moustafa and Jill Slay [11] propose a newer dataset for network traffic that has normal and attack packets, which is known as UNSW-NB15 dataset. This dataset consists of 2,540,047 records, each record represents a network packet described using 47 attributes, in addition to two labels. One label represents the state of the packet, normal or attack packet, while the other label represents the type of attack, out of nine attacks, that the packet is a part of, in case of attack packets. These attacks are the Dos, Exploit, Backdoor, Generic, Fuzzers, Analysis, Reconnaissance, Shellcode and Worms.

3. MACHINE LEARNING TECHNIQUES

Different machine learning techniques are used to implement intrusion detection systems using the datasets illustrated earlier. As the firewalls are the network components that are responsible of analyzing the packet information in order to make a decision of allowing the packet access to the network or denying it, these techniques are used with these firewalls to protect the networks. The information of each packet is retrieved by the firewall and sent to the machine learning technique in order to predict whether it is of a normal or attack traffic. These predictions are used to make and execute the decisions in the firewall. In this section, machine learning techniques employed in intrusion detection systems are illustrated [12].

The *k*-Neared Neighbor (*k*-NN) classifier is a lazy classifier that extracts no knowledge from the training dataset until a prediction is required from the classifiers. This knowledge is extracted every time a prediction is required by retrieving the *k* most similar data objects in the training dataset and select the class dominating these objects. The dominant class can be computed by voting, i.e., the class that has the highest number of data objects, or by weighting each class using the distances between the new object from one side, and each object in the training dataset from another. As the distance between two objects is affected by the number of features that characterize these objects, it is important to reduce the number of features to the minimum possible number, to reduce the time required to compute the distance and accelerate the computations. To do so, each feature is ranked according to its role in the classification process, and features with the lowest ranks are eliminated, or optimization algorithms are used to eliminate any features that do not have any effect, or have a negative effect, on the classification results [13].

Decision tree classifiers use sets of IF/THEN clauses distributed in a tree-like multi-level distribution, so that, the comparison selected from a certain level is decided by the results of the comparison made in the previous level. These sets are generated during the training phase of the classifier and applied to the features values of the new data objects in order to predict a class for each one. Features with higher effect on the class prediction operation are located higher in the decision tree, closer to the root of the tree, wherein decision trees the roots are on the top of the trees [14]. Moreover, some techniques use multiple trees to provide better classification results. Each tree is trained using a different random set of the training dataset. Thus, such classifiers are known as random forest, where the predicted class is the dominant class among those predicted by the trees in the forest [15]. A sample decision tree is shown in Figure 1, to predict the state of a player, whether to play today or not, depending on the weather forecasts.

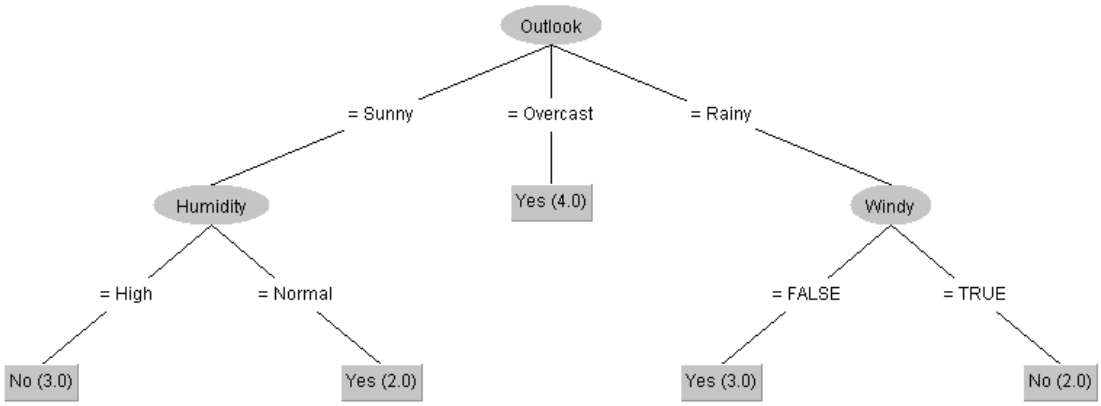

Figure 1. Example of a decision tree based on weather forecast.

Support Vector Machine (SVM) is another popular classifier that is employed in intrusion detection systems. This classifier distributes objects from the training dataset in an *n*-dimensional space, where *n* is equal to the number of features in the dataset. The boundaries among the regions that include objects of the same class are optimized by the SVM classifier, so that, the distances between the boundary and the closest object from each adjacent class is maximized. This ensures higher prediction confidence, as the confidence of SVM's prediction is described by the distance from the new data object that the class is predicted for, and the boundary of that class [16]. An example of the SVM boundaries of a two-dimensional space with three classes is shown in Figure 2.

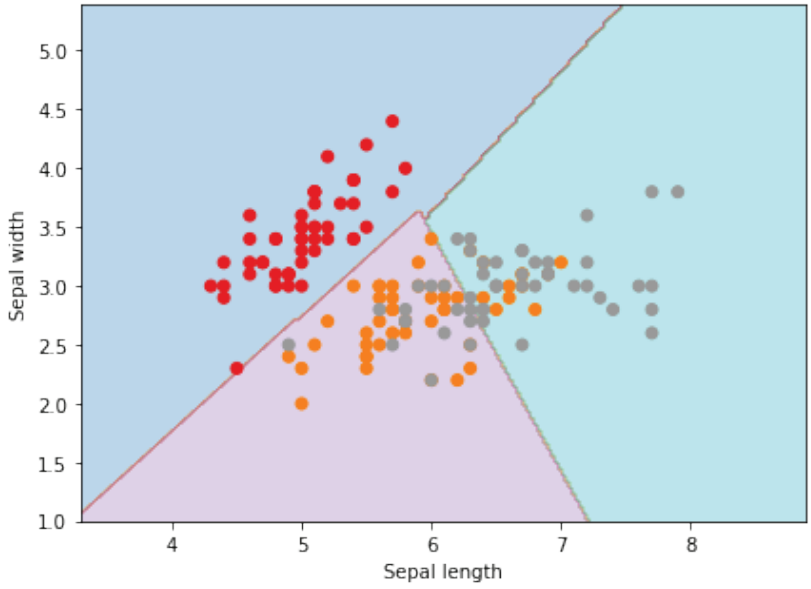

Figure 2. Example of SVM boundaries in a two-dimensional space.

Recently, Artificial Neural Networks (ANN) have gained significant attention among other machine learning techniques, according to the relatively better performance, especially with larger datasets. An artificial neural network is a mathematical representation of the neurons in a human brain and the interconnections among these neurons, which control the decision made by humans. These neurons are distributed in layers, where three types of layers exist in an ANN. The first layer type is the input layer, which has a number of neurons equal to the number of features that characterize the dataset. The second type of layers is the output layer, which has a number of neurons equal to the number of outputs intended from the network. As the topologies of these layers are controlled by the way these networks interface the external domain, a third type of layers is added to these networks, which is the hidden layers, in order to provide more flexibility to the overall topology of the ANN. The number of hidden layers, as well as the number of neurons in each layer is configured according to the needs of the neural network, where the number of neurons in a layer controls the number of features that the layer can detect, while the number of layers controls the complexity of the features that can be detected in these layers [17]. A sample feed-forward neural network is shown in Figure 3.

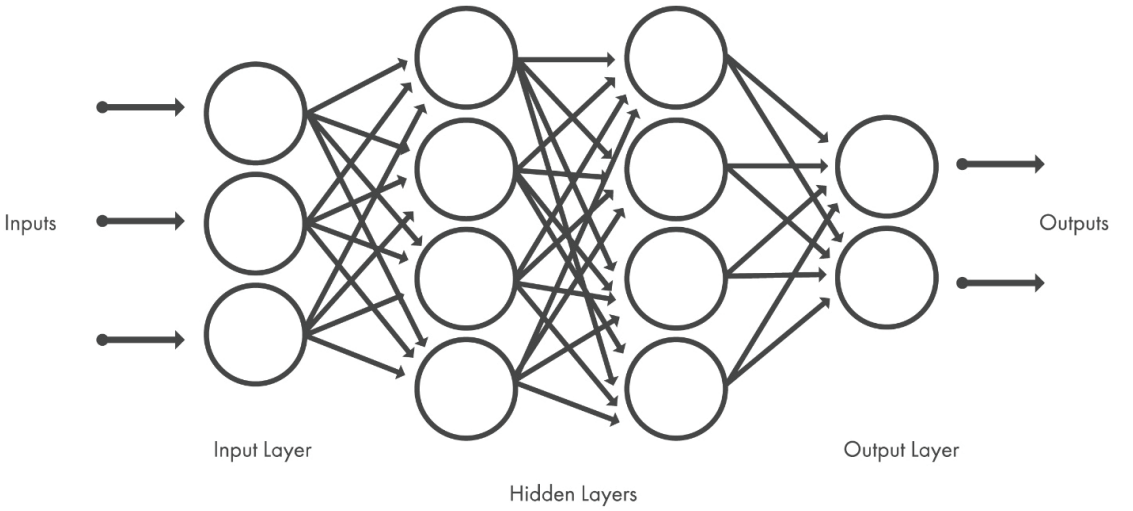

Figure 3. Sample feed-forward artificial neural network.

4. INTRUSION DETECTION SYSTEMS

Many intrusion detection systems are proposed based on machine learning techniques. These systems have shown different performances depending on the dataset, used for the training and evaluation, and the machine learning technique used in the system. The system proposed by Wei-Chao Lin, et al. [18] uses the *k*-NN classifier to predict the state of each network packet, whether to be from a normal or attack traffic. This system is trained and evaluated using the KDD CUP'99 dataset, where the evaluation measures show a good prediction accuracy of 99.89% accurate predictions. However, as the *k*-NN classifier is a lazy classifier, the knowledge is extracted each time a prediction is required, i.e., the training dataset is scanned every time a new packet enters the network, which is a very resource-consuming process

that requires either expensive servers with high resources, or longer execution time that may degrade the quality of the services provided on that network.

Neha G Relan and Dharmaraj R Patil [19] , which perform an intrusion detection system using the decision tree classifier. The performance of the proposed system has scored a highest of 95.09%, using the KDDCUP'99 dataset for both training and testing stage. The decision tree classifier generate sets of IF/THEN rules that can be applied to the attributes' values of each tuple, in order to predict a class for that tuple. These sets are created depend on the attributes values of the tuples in the training dataset, and the label that each record belongs to, where the sets are distributed in levels, and the condition to be investigated in the next level is selected depending on the outcome of the condition being applied in the current level.

Malek Al-Zewairi, et al. [20] suggest an intrusion detection system depend on deep learning that include of five hidden layers with ten neurons in each layer. The deeper the neural network, the more complex attribute can be discover based on the input data, while rising the number of neurons in a layer rising the number of attribuye that the layer can detect. The accuracy of the deep learning model is compared to other classifiers, such as logistic regression, decision tree, Naïve Bayes and neural network, where the experimental results show that the deep learning model has scord the highest withe 98.99% accuracy when tested with the UNSW-NB15 dataset.

5. CONCLUSION

Network attack methods are developing rapidly in order to execute intrusions using network traffic similar to normal traffic, so that, detecting these attacks becomes more difficult using traditional techniques. For this reason, intrusion detection systems are developed to use machine learning techniques to gain the ability of making more complex decision and protect the network from any intrusion attempts. Machine learning techniques have the ability to extract knowledge from a set of inputs collected from the external world. This knowledge is then used to assist making more appropriate decision based on the characteristics of the new data objects fed to the machine learning technique. Classification is one of the widely used supervised data mining techniques, where data mining is the field of machine learning that is concerned with processing datasets. A classifier extracts the characteristics of objects in each class to predict a class for new data objects, depending on their characteristics.

Many datasets are collected for packets in network traffic that contains both normal and attach packets, so that, these datasets can be used to train classifiers on how to detect attack packets incoming to the network, based on their characteristics. These predictions are used to come up with the appropriate decision, whether to allow the packet through to the network, or block it. Different studies are conducted that employ many machine learning techniques in intrusion detection systems. As these techniques have different approaches to extract knowledge from the datasets, they have shown different performance measures depending on the techniques and the training dataset. However, techniques that use Artificial Neural Networks have shown a better overall performance, compared to other techniques used for this purpose.

6. REFERENCES

D. Acemoglu, A. Malekian, and A. Ozdaglar, "Network security and contagion," *Journal of Economic Theory,* vol. 166, pp. 536-585, 2016.

D. Yu, Y. Jin, Y. Zhang, and X. Zheng, "A survey on security issues in services communication of Microservices-enabled fog applications," *Concurrency and Computation: Practice and Experience,* p. e4436.

V. C. Storey and I.-Y. Song, "Big data technologies and Management: What conceptual modeling can do," *Data & Knowledge Engineering,* vol. 108, pp. 50-67, 2017.

I**. H. Witten, E. Frank, M. A. Hall, and C. J. Pal,** *Data Mining: Practical machine learning tools and techniques*: Morgan Kaufmann, 2016.

M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications,* vol. 60, pp. 19-31, 2016.

K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.

K. Cup, "Dataset," *available at the following website http://kdd. ics. uci. edu/databases/kddcup99/kddcup99. html,* vol. 72, 1999.

M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on*, 2009, pp. 1-6.

J. McHugh, "Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory," *ACM Transactions on Information and System Security (TISSEC),* vol. 3, pp. 262-294, 2000.

M. S. Pervez and D. M. Farid, "Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs," in *Software, Knowledge, Information Management and Applications (SKIMA), 2014 8th International Conference on*, 2014, pp. 1-6.

N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Military Communications and Information Systems Conference (MilCIS), 2015*, 2015, pp. 1-6.

J. Suuronen and M. Bergenwall, "System and method of providing virus protection at a gateway," ed: Google Patents, 2016.

Y. Liao and V. R. Vemuri, "Use of k-nearest neighbor classifier for intrusion detection1," *Computers & security,* vol. 21, pp. 439-448, 2002.

J. R. Quinlan, *C4. 5: programs for machine learning*: Elsevier, 2014.

J. Zhang, M. Zulkernine, and A. Haque, "Random-forests-based network intrusion detection systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 38, pp. 649-659, 2008.

J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters,* vol. 9, pp. 293-300, 1999.

M. Kubat, "Artificial neural networks," in *An Introduction to Machine Learning*, ed: Springer, 2015, pp. 91-111.

W.-C. Lin, S.-W. Ke, and C.-F. Tsai, "CANN: An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowledge-based systems,* vol. 78, pp. 13-21, 2015.

N. G. Relan and D. R. Patil, "Implementation of network intrusion detection system using variant of decision tree algorithm," in Nascent Technologies in the Engineering Field (ICNTE), 2015 International Conference on, 2015, pp. 1-5.]

M. Al-Zewairi, S. Almajali, and A. Awajan, "Experimental Evaluation of a Multi-layer Feed-Forward Artificial Neural Network Classifier for Network Intrusion Detection System," in *2017 International Conference on New Trends in Computing Sciences (ICTCS)*, 2017, pp. 167-172.