Large-Scale Network Intrusion Detection

O. Al-Jarrah[1], P. D. Yoo[1,2], K. Kim[2]
[1]ECE Dept., KUSTAR, Abu Dhabi, UAE
[2]CS Dept., KAIST, Daejeon, South Korea

{omar.aljarrah, paul.yoo}@kustar.ac.ae, kkj@kaist.ac.kr

***Abstract*—Intrusion Detection System (IDS) monitors and analyzes networks' activities for potential intrusions and security attacks. However, the performance of existing IDSs does not seem to be satisfactory due to the rapid evolution of sophisticated cyber threats in recent decades. Moreover, the volumes of data to be analyzed are beyond the ability of commonly used computer software and hardware tools. They are not only large in scale but fast in/out in terms of velocity. In large-scale IDS, the one must find an efficient way to reduce the size of data dimensions and volumes. In this paper, we propose novel feature selection methods, namely, RF-FSR and RF-BER. The features selected by the proposed methods were tested and compared with three of the most well-known feature sets in the IDS literature. The experimental results showed that the selected features by the proposed methods effectively improved their detection rate and false-positive rate, achieving 99.8% and 0.001% on well-known KDD-99 dataset, respectively.**

Keywords—large-scale data; intrusion detection system; random forest; styling; features selection.

I. INTRODUCTION

Due to recent technological advances, network-based services have become increasingly vital in modern society. Intruders look for the vulnerabilities of computer systems in order to compromise their communications or to gain illegal access to the core of the systems. However, existing security mechanisms are still inflexible, unscalable, and not powerful enough to deal with such attacks.

In early days of Intrusion Detection Systems (IDS), the rule-based methods were dominant. These methods find the intrusions by comparing its characteristics to known attack signatures. Security experts manage the computer-encoded rules which are extracted from real intrusions. As the network traffic grows rapidly, keeping these rules updated becomes more and more difficult, tedious, and time-consuming. Since then, Machine Learning (ML) based methods were introduced to the problem of network intrusion detection. ML refers to computer algorithms that have ability to learn from past examples. In context of intrusion detection, a detection model learns from previously recorded attack patterns (i.e., signatures), and detects similar ones in incoming traffic. The popularity of ML-based models came from the fact that it could be "tailored" to the network data of the system where the model is being used. ML-based IDSs have performed well in the literature as well as the reality. However, the "model-free" property of such methods causes relatively high-computational cost. Moreover, as the volume and velocity of network data grows rapidly, such computing cost issues must be resolved. Hence, this paper gives an insight into the features selection techniques in IDS and proposes two different novel feature selection methods that could help improve performance of any ML-based IDS. The proposed methods use an ensemble of RF algorithm, with forward and backward ranking features selection techniques [1–2]. To prove the usefulness of the proposed methods, we compare our results with those of other three well-known feature sets [3–5] on KDD-99 dataset.

II. METHODS

Our approach consists of four consecutive steps. i. dataset selection, ii. feature selection, iii. model selection and iv. evaluation.

A. Datasets Selection

Among several publicly available datasets, KDD-99 is the most widely accepted benchmark. However, KDD-99 dataset has some drawbacks [6]. First, the full dataset is large which increases the computational cost of the IDS. Therefore, only 10% of the set is usually used [7]. Second, KDD-99 has many redundant data in the training set and duplicated records in the testing set. That might affect the learning process. It causes learning biasing to the frequent records and prevents learning the infrequent records, which might be more harmful to the system.

NSL-KDD99 dataset is a filtered version of KDD-99. The redundancies between testing and training sets have been minimized. Due to its reduced size, a learning algorithm could learn from NSL-KDD99 most instantly [6–7]. Therefore, IDSs can use the whole dataset while detecting different types of attacks more precisely. Similar to KDD-99, NSL-KDD99 has 41 features that contain both normal and attack patterns.

B. Features Selection

Feature selection refers to the process of selecting a subset of relevant features that fully describes the given problem with a minimum degradation of performance [8]. Finding right features has a significant impact on IDS's performance as it reduces the computation cost, removes information redundancy, increases the accuracy of detection algorithm, facilitates data understanding and improves generalization [9].

Machine-learning algorithms were used in feature selection process. In [1], Random Forest (RF) was used to sort all the 41 features according to their weight. While, Enhanced Support Vector Decision Function (ESVDF) method was proposed in [2]. ESVDF method uses Support Vector Decision Function (SVDF) to find different features weight. Then, features

correlations, which are the dependence of features on each other, are determined by either Forward Selection Ranking (FSR) or Backward Elimination Ranking (BER) algorithms[2].

Kaycik in [3] proposed a feature selection method. The proposed method analysis KDD'99 dataset and its 41 features, based on calculating the information gain and the entropy of each feature to measure its relevance [3]. The last features set, which is used for comparison in this paper, are the 6 important features. These 6 features are chosen by experts as representatives for the 41 features [5].

C. Model Selection

Single classifiers, hybrid classifier, and ensemble classifier are examples of ML models that used in IDSs. Single classifier uses one machine-learning algorithm to classify data. Ensemble classifier combines multiple models to generate a single model that has better prediction accuracy. Random Forest (RF) is an ensemble which consists of crowd of decision trees, each one of the decision trees gives a classification of the input data. After that, voting takes place and the forest comes up with the final classification decision based on the voting result.

D. Evaluation

Several experimental settings were considered to evaluate the performance of each features set. These include detection rate (also known as sensitivity-Sn), accuracy (Acc), training time (Tr), Mathew's correlation coefficient (Mcc), and, False Alarm Rate (Far). We aim to have a high Acc, Sn, and Mcc while low in Tr and Far. Some of these parameters are defined as follows:

$$Far = \frac{FP}{TN+FP},$$
$$Acc = \frac{TP+TN}{TP+TN+FP+FN},$$
$$Sn = \frac{TP}{TP+FN},$$
$$Mcc = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}.$$

III. EXPERIMENTS AND RESULTS

The two proposed feature selection methods are devised by combining the results of researches [1–2] in a way that the weight of the features is determined from research [1] using the RF approach, then both FSR and BER are applied separately. The third feature selection method that is used to compare the proposed methods is done by Kaycik [3]. The fourth feature selection method is proposed by [4]. In this study, a hybrid approach is used to obtain the optimal set of 14 features. Table 1 shows the features sets used in this study. NSL-KDD99 dataset is used for training and testing each selected features. 10 fold cross-validation technique was used to perform the experiment.

TABLE 1: FEATURE SETS

Method	Features
RF-FSR	1, 3, 4, 5, 6, 8, 13, 16, 10, 23, 24, 32, 33, 35, 36
RF-BER	1, 2, 3, 5, 6, 10, 14, 16, 32, 33, 36, 37, 38, 41
Kaycik [3]	1, 2, 3, 4, 5, 6, 8, 11, 12, 16, 23, 24, 26, 32, 33
Araújo [4]	2, 3, 5, 6, 9, 11, 12, 14, 22, 30,31, 32, 35, 37
Kantor [5]	1, 2, 3, 4, 5, 6
KDD-99	1–41

Both KDD-99 and NSL-KDD99 datasets have the identical 41 features.

TABLE 2: EXPERIMENTS RESULTS

Method	Tr	Sn (DR)	Acc	Mcc	Far
RF/FSR	12.75	**99.857**	**99.901**	**0.99801**	0.000609
RF/BER	11.52	99.833	99.881	0.99761	0.000772
Kaycik [3]	9.76	99.732	99.809	0.99616	0.001247
Araújo [4]	12.23	99.840	99.891	0.99781	0.000639
Kantor [5]	**4.77**	99.499	99.354	0.98702	0.007722
KDD-99	22.09	99.830	99.895	0.99790	**0.000505**

As seen in Table 2, the proposed features selection method RF-FSR achieved best performance measures in terms of Sn (Detection Rate), Acc and Mcc. Due to low number of features, Kantor's feature set obtained the least Tr. For the same reason, full features set KDD-99 achieved the highest Tr and the best Far results.

IV. CONCLUSION

In this paper, we presented two features selection methods, namely, RF-FSR and RF-BER, the novel ensembles of decision-tree-based (J48/C4.8) voting algorithm with forward selection / backward elimination feature raking techniques. Such feature selection method is of great importance, especially for an IDS designed for large-scale networks where the volume and velocity are high. In this paper, the features selected by the proposed methods were compared with other three popular feature sets on widely known KDD-99 and NSL-KDD99 datasets. The experimental results showed that the feature set selected by our proposed RF-FSR technique outperformed all other well-known feature sets in the literature, which seems to be promising and suitable for large-scale network IDSs.

ACKNOWLEDGMENT

This research was supported by the KUSTAR-KAIST Institute, under the R&D program supervised by the Korea Advanced Institute of Science and Technology (KAIST), South Korea.

REFERENCES

[1] ENGEN, "Machine learning for network based intrusion detection," Doctoral dissertation, Bournemouth University, 2010.

[2] S. Zaman and F. Karray, "Features selection for intrusion detection systems based on support vector machines," in Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE, pp. 1–8, 2009.

[3] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "Selecting features for intrusion detection: a feature relevance analysis on kdd99 intrusion detection datasets," in Proceedings of the third annual conference on privacy, security and trust, Citeseer, 2005.

[4] N. Araujo, R. de Oliveira, E.-W. Ferreira, A. Shinoda, and B. Bhargava, "Identifying important characteristics in the kdd99 intrusion detection dataset by feature selection using a hybrid approach," in IEEE 17th International Conference on Telecommunications (ICT), pp. 552–558, IEEE, 2010.

[5] P. Kantor, G. Muresan, F. Roberts, et.al. "Analysis of three intrusion detection system benchmark datasets using machine learning algorithms," in Intelligence and Security Informatics, Germany, Berlin Heidelberg, sec. 3, pp. 363 Springer - Verlag, 2005.

[6] M. Tavallaee, E. Bagheri, W. Lu, and A.-A. Ghorbani, "A Detailed analysis of the kdd cup 99 data set," in Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications 2009, 2009.

[7] Y.-X. Meng, "The practice on using machine learning for network anomaly intrusion detection," in International Conference on Machine Learning and Cybernetics (ICMLC), vol. 2, pp. 576–581, IEEE, 2011.

[8] T. Lappas and K. Pelechrinis, "Data mining techniques for (network) intrusion detection systems," Department of Computer Science and Engineering UC Riverside, Riverside CA, vol. 92521, 2007.

[9] M. Bhuyan, D. Bhattacharyya, and J. Kalita, "Network anomaly detection: methods, systems and tools," Communications Surveys & Tutorials, IEEE, vol.PP, no.99, pp.1, 34, 0 2013.