# Three Kinds of Network Security Situation Awareness Model Based on Big Data

Bowen Zhu, Yonghong Chen, Yiqiao Cai (Corresponding author: Bowen Zhu)

College of Computer Science and Technology, Huaqiao University Xiamen, 361021, China (Email: 1511314013@hqu.edu.cn) (Received June 7, 2017; revised and accepted Sept. 12, 2017)

## Abstract

In this paper, we have proposed three kinds of network security situation awareness (NSSA) models. In the era of big data, the traditional NSSA methods cannot analyze the problem effectively. Therefore, the three models are designed for big data. The structure of these models are very large, and they are integrated into the distributed platform. Each model includes three modules: network security situation detection (NSSD), network security situation understanding (NSSU), and network security situation projection (NSSP). Each module comprises different machine learning algorithms to realize different functions. We conducted a comprehensive study of the safety of these models. Three models compared with each other. The experimental results show that these models can improve the efficiency and accuracy of data processing when dealing with different problems. Each model has its own advantages and disadvantages.

Keywords: Big Data; Machine Learning; Network Security Situation Awareness

## 1 Introduction

Big data has become a hot topic in recent years. Many of this dataset is generated in the network environment, its characteristics are a large size and high dimension. It is a huge challenge for knowledge discovery, such as network traffic anomalies. At the same time, the research and application of NSSA have gained wider attention as the Internet security has become more important [9].

The scale and topology are expanding and complicated, with the development of the Internet infrastructure. It makes the various threat in the network more subtle. The researchers hope to use NSSA to detect cyber-attacks from a large number of high-dimensional data which has a large amount of noise. Then they can understand the security trends of the whole network from a macro perspective [4, 6, 11]. The situation refers to the synthesis of

each object, which is a holistic and global concept. NSSA refers to understanding the meaning of these elements in a given time and space and to predict the possible effects. Therefore, NSSA is a cognitive process of the network security. It is generally believed that NSSA is comprised of three modulesNSSD, NSSU, and NSSP [4]. The NSSA model is shown in Figure 1. There are many problems that need to be solved. Such as low accuracy, poor forecasting accuracy, poor evaluation, poor performance and low efficiency, etc.



Figure 1: Network security situation awareness model

Abawajy et al. [1] studied the Large Iterative Multitier Ensemble (LIME) classifiers designed specifically for big data security. The classifier uses many basic classification algorithms as the basis for an iteration to form a higher-level classifier to solve big data security issues. By reference to LIME classifier, it is not difficult to find that the algorithm plays a key role in big data analysis. The LIME classifier provides a good module fusion strategy based on a variety of algorithms to efficiently solve big data issues in NSSA. The three NSSA models based on big data are implemented under the guidance of LIME classifier. Therefore, each model combines the data pre-processing function in NAAD and analyzes the association rules based on the dataset in NSSU to improve the accuracy of NSSP. And the parallel experiment of each model is implemented on a distributed platform which improves the efficiency of NSSA.

The task of the first module is to identify all activi-



Figure 2: N-NSSA model

ties in the system and the feature of these activities. It is comprised of data preprocessing and activity modeling. The network data has high dimension and large size. Therefore, the proposed module adopts data normalization and dimensionality reduction methods based on feature decomposition in data preprocessing [10]. Usually, the number of rows in network data is much larger than columns. So we need to limit the number of characteristics within a range to reduce the dimensions of the data, which makes the feature more obvious. Currently, the focus of activity modeling is divided into methods with expertise-based and without prior knowledge. In this paper, the latter is used for activity modeling which based on the clustering algorithms (which can group according to the similarity and dissimilarity) [7].

The task of the second module is to analyze the semantics and relation of network activities to infer the intent of an attacker and anticipate possible attacks. The module adopts the association rule mining algorithm [12], which mainly analyzes the logical relationship between attacks (or multiple recurring patterns and concurrency relation). And then deduces the possible changes of attack. In general, the entire dataset needs to be scanned cyclically when the association rules are analyzed. As the data grows, the cost of analysis will increase geometrically and the cost is unbearable when faced with big data. This module uses a parallel mining method which only requires scanning the dataset twice. On the basis of the scan, each node in the parallel platform performs the association rule analysis and summarizes the relevance of the dataset [13].

The task of the third module is to assess the damage condition which has occurred in the network and make a prediction about the potential threat. Each model will be described in detail in Section 3, and then the experimental results will be analyzed in Section 4.

In this paper, we proposed three NSSA models: Net-

work Security Situation Awareness Model Based on Neural Network (N - NSSA model), Network Security Situation Awareness Model Based on Random Forest (F -NSSA model) and Network Security Situation Awareness Model Based on Star Structure (S - NSSA model) [8,14, 15]. These models analyze the various dangerous signals that exist in the data based on knowledge reasoning. Each model classifies these threats and projects the results of the situation to the actual network environment. More specifically, the contribution of this paper is summarized as follows:

- In view of the shortcomings of the existing models, three novel models are proposed according to the idea of LIME classifier.
- According to the feature of the three models, the advantages and disadvantages of the models are analyzed.
- Experiments on distributed parallel platforms demonstrate the availability and effectiveness of the three models.

## 2 Network Security Situation Awareness Model

In this section, we will introduce the structure and implementation of the three models. And then explain the advantages and disadvantages of these models.

#### 2.1 N - NSSA Model

As shown in Figure 2, this is the N-NSSA model, the model combined with a three-tier feed forward neural network. In the input layer of the neural network, it contains the first and second modules of NSSA. In the hidden layer, it will integrate the results of NSSU and transmit them to output layer to adjust the error in the neural network and make the situation projection.

The N - NSSA model is a back propagation network. On the direction of data transmission, it is not only from the input layer to hidden layer to output layer but also the feedback from the output layer to the input layer. This structure makes the model has a self - learning function which can change the behavior according to the feature of the input data. This characteristic makes the model better able to classify the untrained pattern. And it also can effectively detect the nonlinearity inherent rules of data. The model has a complex structure, so it is not sensitive to some of the outliers in the data which makes the model better able to tolerate noisy data.

Although the N-NSSA model which incorporated into the neural network has the above advantages, there are also some disadvantages. The neural network requires a relatively long time to train the model, especially in the face of big data. And the neural network is more sensitive to missing values and therefore require appropriate data preprocessing. The powerful learning ability of neural network makes N-NSSA model prone to over fitting.

### 2.2 F - NSSA Model

As shown in Figure 3, this is the F-NSSA model which consists of multiple decision trees. Its output result is determined by the number of output results of all decision trees. The structure is divided into three layers from top to bottom. We can get the final result of NSSA at the leaf nodes. The three modules of NSSA converged in this three-tier structure. The NSSD of the first module is performed at the top root node. Its result is transmitted to the second module for NSSU. Finally, the NSSP of the third module is performed at each leaf node.



Figure 3: F-NSSA model

In this model, the input data will be divided into smaller parts. These small parts build tree roots, form branches and a number of leaf nodes (each node represents a conclusion). A path from the root of a decision

tree to a leaf node forms a category prediction of these objects in processed data. The model uses a top-down greedy strategy when building a decision tree. It selects the best-performing attributes at each node to classify processed data and repeats the process until the tree is able to classify these data accurately or all attributes are used.

The building process of each tree in the model is relatively fast and there is no special requirement for the distribution of processed data. There is no requirement and restriction on the pre-processing of the data and there is a high tolerance for the missing values. The model is not susceptible to extreme values and it can be used to deal with both linear and nonlinear relationship in processed data. In the third module, the model summarizes the conclusions of leaf nodes which generated by each decision tree and outputs the results under the majority rule.

There are also some disadvantages of the F-NSSA model. In the process of building a decision tree, the model uses the greedy strategy which seems to make the current best choice, but not from the overall consideration. So it is easy to have a locally optimal choice. At the same time, the model lacks a variety of evaluation methods and does not suitable for continuous variables. In the case of an excessive number of variables, there will be the risk of over fitting.

## 2.3 S - NSSA Model

As shown in Figure 4, this is the S-NSSA model which is implemented by reference to the star topology. It can be divided into two parts: the peripheral part (which is divided into N nodes according to processed data) and a core part. The peripheral part contains two modules (NSSD & NSSU) and the results of NSSU will be transmitted to the core part. The core part of the S-NSSA model is based on the Naive Bayesian algorithm. Bayesian is a very mature statistical classification method, it is mainly used to predict the possibility of a relationship between members of the class (For example, the probability of a given category is determined by the properties of a given observation value). The S-NSSA model collects the results of each node in the peripheral part to understand the results. The model gets the overall situation through data fusion.

In the S-NSSA model, it is less sensitive to missing value due to the advantages of the Naive Bayesian algorithm. And the algorithm is simple, so the efficiency of classification is stable. In the face of the small-scale dataset, the model has a very good performance. It can handle multi-classification tasks. When dealing with big data parallelization is a good choice. It is not difficult to find that the main difficulty in estimating the posterior probability based on the Bayes'theorem is that the class conditional probability is the joint probability on all attributes and it is difficult to obtain directly from the limited train-set. In order to avoid this obstacle, the tra-



Figure 4: S-NSSA model

ditional Naive Bayesian Classifier takes the assumption that all attributes are independent of each other (each attribute of processed data affects the classification result independently). But this is unavoidable in the actual processing of the data.

## 2.4 Complexity Analysis

In this subsection, a theoretical analysis is conducted to access the computational complexity of the three NSSA models. The efficiency of these models will be affected by the computer hardware, software and the scale of the cluster. These factors will mask the merits of these models. So it is assumed that the time complexity of these models is related to the scale of the issue. Each model is divided into three modules and their complexity determines the complexity of each model. First, we define several symbols for subsequent analysis. N: the number of objects to be processed, K: the number of categories contained in the data, t: the number of iterations in the process, and d: the dimension of the data.

In the NSSD module, the data preprocessing operation is carried out. The dimension reduction is the generation of more obvious data from a large number of high dimensional data and its complexity is o(nlogd \* t + nt). The data is classified according to the characteristics of the data. The analysis process is mainly based on the distance between the data, and its complexity is o(ntk). When parallel operated in the distributed platform, it is calculated by multiple nodes in the cluster at the same time, so k and t can be considered a constant, so the time complexity is o(n). The correlation analysis of the data is carried out in the NSSU module. In the process of analysis, an optimized strategy is used to analyze the data according to the attributes and these attributes are analyzed on each tree, so its complexity is o(lognd).

In the N-NSSA model, the neural network is a main structure of the model and the other modules are included. So the complexity of this model is mainly determined by these process. The time complexity of the N-NSSA model is o(n (logd + 1) t + lognd + 1) in the process of self-learning stage for error backpropagation. The forest is the main part of the F-NSSA model and the consumption of each node is the process of building a tree. The complexity is mainly related to the dimension of the data and the amount of data. So the complexity is o(nd). So the complexity of the F-NSSA model is  $o(\log + 1)$  t + nd). The S-NSSA model is divided into two part. The peripheral part is distributed in each node, its complexity is o(nt (logd + 1) + lognd). The core part uses the Bayes'theorem. Its complexity is mainly related to the size of the data, so the time complexity of the model is: o(nt (logd + 1) + lognd) + n).

## 3 Experimental Results and Analysis

Wu et al. [12] argue that the challenges of big data mining are divided into three levels. One of them is the challenge of the data mining platform. Due to a large amount of data, big data processing requires the use of parallel computing architectures. One of the major ways to deal with big data depends on the Hadoop platform [2]. The computational framework used in this paper is MapReduce in the Hadoop ecosystem which is a batch parallel processing computational framework with many machine learning and data mining algorithms. Using the computational framework to derive the relation between processed data from a large number of historical data. On this basis to predict the next action of the attacker accurately [3].

Our experiments were designed to evaluate the NSSA of the three models. It is necessary to evaluate the three models proposed in this paper. The performance of each model cannot depend only on theoretical analysis. The results of these experiments shown below will help further study. Each model has advantages and disadvantages. The model performance was tested in the 1999 KDD - cup dataset and the 2015 CAIDA dataset. Three sets of experiments were conducted in this paper and each set was divided into two or three parts [5].

### 3.1 Comparison of True Positive Rate

The first set of experiments was divided into three parts. The experiment was carried out on the 1999 KDD-CUP dataset. The dataset defines a network connection record as a sequence of TCP packets from start to end in a certain period of time and during this time the data is transmitted from the source IP address to the destination IP address under a predefined protocol (such as TCP or UDP). Each network connection record is marked as normal or anomaly and the abnormal type is subdivided into four major categories. There are 39 types of attacks in over 90%. In this part of the experiment, the true positive the dataset, 22 types are in the training set and the rest are in the test set. The same dataset is used in the same set of experiments.



Figure 5: Comparison of true positive rate

The first part of the experiment compares the true positive rate of the third module of each model. The NSSP module is implemented by the core algorithm of the model (naive Bayesian, random forest, neural network). The experimental results are shown in Figure 5. From the figure, we can see that the true positive rate of F-NSSA model is better than the other two models and the N-NSSA model is the worst. Because the primary data is not preprocessed. There are extreme values and noise in the data. The F-NSSA model has no requirement for the distribution of the data and it has a good tolerance to the missing values. It is not easily affected by the extreme values, so the F-NSSA model is better.

In the second part of the experiment, we compare the first and third modules. The primary data is preprocessed in the first module. The dimensions of primary data are high and it contains noise. These data is normalized and reduced which is beneficial for subsequent analysis after preprocessing. And then analyzing the relationship between these data to modeling activities (identify activities and extract features through clustering). This makes the characteristics of each category of experimental data more obvious. And then the results of NSSD will be transmitted to the third module (NSSP). Comparing the true positive rate of each model. The experimental results shown in Figure 5, we can see from the figure that the true positive rate of this experiment is improved and the F-NSSA model is still the best.

The third part of the experiment includes all the modules of the model. The NSSU module analyzes the logical relation between the anomalies. Finding the association rules between each anomaly that is hidden in the data. And infer the possible changes in the anomaly. Understanding the meaning of the anomaly and transmitting the results of NSSU to NSSP to do the final judgment of NSSA. From the experimental results in Figure 5, we can see that the true positive rate of the three models proposed in this paper is much higher and the accuracy rate is

rate of N-NSSA model and S-NSSA model exceeds the F-NSSA model. After preprocessing the primary data, the N-NSSA model and S-NSSA model overcame the sensitivity of the dataset, thus the true positive rate was higher.

#### 3.2**True Positive Rate and False Positive** Rate

The second set of experiments was divided into three parts which use the 1999 KDD-CUP dataset. The dataset is divided into four anomalies (DOS, R2L, U2R, PROBING). Each anomaly contains a number of attack types. In the experiment, we re-classify all the attack types in 4 and then add a large amount of normal data to each class to simulate a real network environment. The first part of the experiment uses the N-NSSA model to verify the true positive rate and false positive rate. The second part of the experiment uses the F-NSSA model to verify the true positive rate and false positive rate. The third part of the experiment uses the S-NSSA model to verify the true positive rate and false positive rate. The experimental results are shown in Figure 6. We evaluate the performance of each model through two evaluation indicators. The first indicator is the true positive rate. The second indicator is the false positive rate. From the figure, we can see that the true positive rate of each model is more than 90%, and the false positive rate is less than 10%. They are able to detect each anomaly well. So the three models for network security situational awareness can have good performance.



Figure 6: Comparison of TP & FP

#### 3.3Size-up and Speed-up

The third set of experiments is divided into two parts. The experimental data is based on the CAIDA dataset. The dataset contains passive detection Internet anonymous data. The size of the dataset reaches to TB level. This set of experiments uses the dataset about 20%(20GB).

The first part of the experiment is the time-efficiency comparison of the three models. In the case of the same node (10 nodes) in the Hadoop cluster to process the dataset with different size. The experimental result is shown in Figure 7. In this part of the experiment, seven sizes of the dataset are divided (100MB, 500MB, 1GB, 2GB, 4GB, 8GB, and 16GB). From the curve, in the figure, we can see that the three models are relatively stable when dealing with big data.



Figure 7: Size-up

In the figure, we can get this conclusion. The N-NSSA model always consumes the most time when dealing with the same size of the dataset. The S-NSSA model is followed. The F-NSSA model is most efficient. Because the N-NSSA model contains a self-learning stage for error backpropagation which requires constantly learning to adjust the error of judgment. So that can improve the accuracy of NSSA model. This process sacrifices some time but improves the accuracy. The third part of the first set of the experiment can prove it. From the perspective of a structural feature of the S-NSSA model. Although the peripheral module is parallelized at the same time by many nodes, all the data in the NSSP module is processed through the central core part. This leads to a poor performance in terms of time efficient than the F-NSSA model. The F-NSSA model divides a large amount of data into relatively small units and then processes a relatively small portion of the data at each node. Each node builds one decision tree which constitutes the entire network security situation. The final result is judged by each node which avoids one-sidedness and makes it very efficient when dealing with big data.

The second part of the experiment is the processing time comparison between the three models. In the Hadoop cluster, the number of nodes increases gradually when the amount of data is constant. The experimental result is shown in Figure 8. With the expansion of the cluster, the communication and transmission consumption between each node increases. It can be seen from the figure that the acceleration ratio in the N-NSSA model is low. Because the consumption between the nodes is large during the error adjustment stage of the model. In this part of the experiment, the F-NSSA model and the S-NSSA model has the similar acceleration ratio. As we have already mentioned, the structural features of the F-

NSSA model make it relatively fewer data transmitted between each node in the process of NSSA and the acceleration ratio curve is approximately linear. The first two module of the S-NSSA model is the same as the F-NSSA model. Each node independently processes the data so that it has a good parallel effect. However, there is a lot of data transmitted between all the nodes in the third module. So the acceleration ratio decreases as the number of nodes increases.



Figure 8: Speed-up

According to the experiments, we can draw the following conclusions. Firstly, we have a higher demand for the accuracy of NSSA, but the rules between the dataset are not easy to mining. And it is not sensitive to the time efficiency. The N-NSSA model is more competent. The accuracy of the N-NSSA model will increase with iteration. However, we should pay attention to the size of the training set to avoid the over-fitting situation. Secondly, when the size of data is very large and contains a lot of extreme values or noise. The F-NSSA model is more appropriate because the structural feature of the model makes it less sensitive to data distribution and easier to handle big data. Finally, when the first two models are not able to adapt to the situation, the S-NSSA model is a good choice. Due to the stability of the model, it makes the true positive rate is better and the parallel processing of peripheral part of the model makes time efficiency can also be accepted. So it is better to choose a targeted model when confronted specific data and different requirements.

## 4 Conclusions

This paper introduces and studies three kinds of NSSA model. These models have been implemented on the distributed platform and achieved a good experimental result. And we describe the composition of each model. These models can deal with different issues. The S-NSSA model has a performance bottleneck. It is not difficult to find out from the experiment that the acceleration ratio of the S-NSSA model decreases with the increase of nodes. From this point, the other two models can better handle multi-source heterogeneous data. The error

backpropagation algorithm based on the neural network can improve the accuracy of N-NSSA model by continuous learning which is a great advantage of the model. The tree-building process can well integrate with the distributed platform, so the F-NSSA model has high speed in the face of big data.

We conducted a systematic scientific experiment. The experimental results show that the existing machine learning and data mining algorithms are effective. When parallelizing these algorithms on the Hadoop platform to deal with big data. This gives us a new idea to study and deal with new issues brought by big data. When the standalone cannot solve these issues, we can solve it by calling the parallelized algorithm of the iterative fusion.

## Acknowledgments

Above work is supported by National Science Foundation (NSF) of China under grant Nos.61370007, 61572206, UI405254. Fujian Provincial Natural Science Foundation of China under grant No.2013J01241, and Program for New Century Excellent Talents of Fujian Provincial under grant No.2014FJ-NCET-ZR06. Huaqiao University graduate research innovation ability cultivation project of China under No, 1511314013.

## References

- J. H. Abawajy, A. Kelarev, and M. Chowdhury, "Large iterative multitier ensemble classifiers for security of big data," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 3, pp. 352–363, 2014.
- [2] M. R. Ahmadi, "An intrusion prediction technique based on co-evolutionary immune system for network security (CoCo-IDP)," *International Journal of Net*work Security, vol. 9, no. 3, pp. 290–300, 2009.
- [3] T. Bhaskar, K. B. Narasimha, and S. D. Moitra, "A hybrid model for network security systems: Integrating intrusion detection system with survivability," *International Journal of Network Security*, vol. 7, no. 2, pp. 249–260, 2008.
- [4] S. Qi, G. Jian, X. D. Zang, "Survey of network security situation awareness," *Journal of Software*, vol. 11, no. 23, pp. 1–17, 2016.
- [5] S. M. Hashemi and J. He, "An evolutionary multiobjective approach for modelling network security," *International Journal of Network Security*, vol. 19, no. 4, pp. 528–536, 2017.
- [6] S. Islam, H. Ali, A. Habib, N. Nobi, M. Alam, and D. Hossain, "Threat minimization by design and deployment of secured networking model," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 135–144, 2018.
- [7] R. Katipally, L. Yang, and A. Liu, "Attacker behavior analysis in multi-stage attack detection system," in *The Workshop on Cyber Security & Information Intelligence Research*, pp. 1–1, 2011.

- [8] V. D. Katkar and S. V. Kulkarni, "A novel parallel implementation of naive bayesian classifier for big data," in *International Conference on Green Computing, Communication and Conservation of Energy*, pp. 847–852, 2014.
- [9] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.
- [10] Y. Liu, Z. L. Sun, Y. P. Wang, and L. Shang, "An eigen decomposition based rank parameter selection approach for the nrsfm algorithm," *Neurocomputing*, vol. 198, no. C, pp. 109–113, 2016.
- [11] E. U. Opara and O. A. Soluade, "Straddling the next cyber frontier: The empirical analysis on network security, exploits, and vulnerabilities," *International Journal of Electronics & Information Engineering*, vol. 3, no. 1, pp. 10–18, 2015.
- [12] X. Wu, X. Zhu, G. Q. Wu, and W. Ding, "Data mining with big data," *IEEE Transactions on Knowledge & Data Engineering*, vol. 26, no. 1, pp. 97–107, 2014.
- [13] Y. Xie, Y. Ma, C. Ling, and G. Wang, "A novel parallel clustering algorithm PXM based on FP-Tree," in International Symposium on Instrumentation & Measurement, Sensor Network and Automation, pp. 475–481, 2012.
- [14] C. Zhang and D. Yuan, "Fast fine-grained air quality index level prediction using random forest algorithm on cluster computing of spark," in Ubiquitous Intelligence and Computing and IEEE International Conference on Autonomic and Trusted Computing and IEEE International Conference on Scalable Computing and Communications and ITS Associated Workshops, pp. 929–934, 2016.
- [15] C. Zhu and R. Rao, "The improved bp algorithm based on mapreduce and genetic algorithm," in *In*ternational Conference on Computer Science & Service System, pp. 1567–1570, 2012.

## Biography

**Bowen Zhu** Graduate student. His main research direction includes network security and intelligence algorithm.

**Yonghong Chen** Professor. Mainly engaged in computer network and information security research, including Internet of things and security, cloud computing and security, intrusion detection, digital watermarking, big data security.

**Yiqiao Cai** Ph. D. Mainly engaged in intelligent algorithms and their applications, data mining and other aspects of research.