IMPROVEMENT OF FALSE REPORT DETECTION PERFORMANCE BASED ON INVALID DATA DETECTION USING NEURAL NETWORK IN WSN

Sanghyeok Lim and Taeho Cho

Department of Electrical and Computer Engineering, Sungkyunkwan University,
Republic of Korea

ABSTRACT

WSN consists of a number of nodes and base stations and is used for event monitoring in various fields such as war situations, forest fires, and home networks. WSN sensor nodes are placed in fields that are difficult for users to manage. It is therefore vulnerable to attackers, and attackers can use false nodes or MAC injection attacks through the hijacked nodes to reduce the lifetime of the network or trigger false alarms. In order to prevent such attacks, several security protocols have been proposed, and all of them have been subjected to MAC-dependent validation, making it impossible to defend against false report attacks in extreme attack circumstances. As attacks have recently become more diverse and more intelligent, WSNs require intelligent methods of security. Based on the report information gathered from the base station, the proposed method provides a technique to prevent attacks that may occur where all MAC information is damaged by carrying out verification of a false report attack through the machine learning based prediction model and the evaluation function.

KEYWORDS

Network Protocols, Wireless Sensor Network, simulation, machine learning, neural network

1. INTRODUCTION

The wireless sensor network (WSN) consists of sensor nodes and a base station (BS) [1], [2]. It is installed and used in various situations, including war situations, home networks, and forest fire monitoring. Sensor nodes are often placed in hard-to-manage locations and are equipped with low-performance CPUs and low-capacity batteries to save network cost. A network attacker who exploits such vulnerability could launch various kinds of attacks such as false report injection attacks or false Message Authentication Code (MAC) injection attacks by compromising nodes. These attacks cause unnecessary energy consumption of the node or cause false alarms at the BS by sending a report about events that did not actually occur [3],[4],[5].Currently, several security protocols are proposed to prevent false report injection attacks. These protocols determine whether the report is authentic or not depending on whether the MAC included in the report is normal or abnormal [6], [7],[8],[9].Even if filtering is not properly performed, the BS may perform the final MAC check. However, in the case of a report that is generated in a cluster where a large number of nodes are compromised, all MACs from the nodes participating in the report generation will have normal values and the report is securely transmitted to the BS. The final MAC check will also consider if it is a legitimate report. The BS sounds a false alarm and responds abnormally, which leads to a high risk. Attacks are becoming increasingly diverse and difficult to detect, and there are security vulnerabilities that rely solely on MACs. As a result, a more intelligent method of security is needed. The proposed scheme is a security model for a false report attack situation in a WSN environment used in forest fire detection with an event

DOI: 10.5121/ijcnc.2018.10602

prediction model and an evaluation function using a neural network model learned based on report contents. A lot of data is needed to determine the authenticity of the report through the contents received from the nodes. As it is not easy to receive data from the actual field, we use the forest fire state variation model to extract the learning data through the simulation and use it to teach the neural network. Section 2 introduces related research, Section 3 introduces the proposed scheme with an attack scenario, and Section 4 explains the results of the experiment and evaluation of the function's validity. Finally, Section 5 outlines the conclusions.

2. RELATED WORK

2.1. PVFS

The probabilistic voting-based filtering scheme (PVFS) is a cluster-based application layer security protocol [6]. All member nodes constituting the cluster are assigned a predetermined number in the node placement step, and each node is assigned one key.

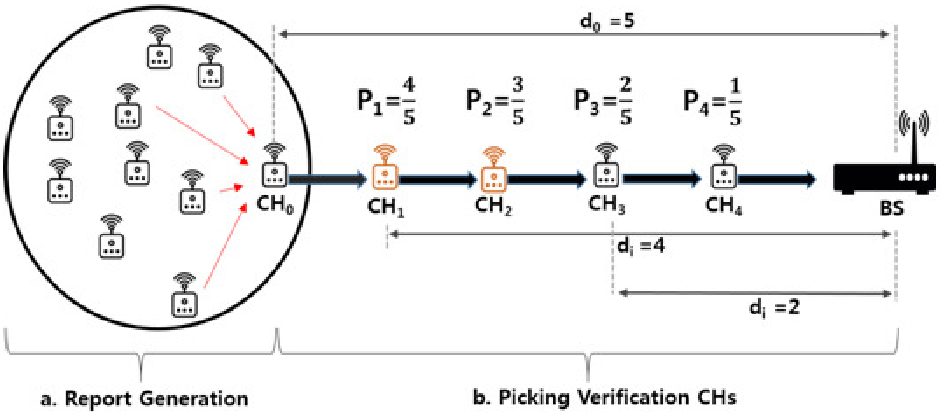

Figure 1. PVFS en-route filtering

When an event occurs, the cluster head node (CH) generates a report, and the member node verifies the contents of the report. If the report is validated and found to be valid, the member nodes attach the MAC generated by their own key to the report. This MAC is used for verifying the authenticity of the report in the next verification node. The verification node is probabilistically selected based on the number of hops between the event detection cluster and the BS, as well as the number of hops between the node and BS. When the verification node receives the report, it compares its own key index with the key index attached to the report. If the key index overlaps, the MAC check is performed. If the index does not overlap, the report is not verified and it is sent to the node on the next path. If a normal MAC is detected as a result of the check, a normal vote is cast. If a false MAC is detected, a false vote is cast. If the number of votes reaches a preset threshold value, the report is considered a false report and dropped immediately. On the contrary, if a true vote reaches the true threshold value, report is then transmitted to the BS without any further verification steps. Since the BS has all the key information, if the validation nodes of the PVFS are not properly verified, the BS finally verifies the report. If at least one MAC is abnormal in the report arriving at the BS, it is regarded as a false report and discarded. If all MACs are normal, the report is determined a normal report and the BS responds as described in the report.

2.2. Artificial Neural Network

A neural network in machine learning is a nonlinear model used to solve prediction problems in data with complex structures. This data mining classification technique is for finding a certain rule or pattern in a large amount of data [10-14].This method has the disadvantage that it is difficult to explain the relationship between the target variables of the input variables, but it has the advantage that the prediction power is very high. The neural network model contains a component called a hidden node, which is a model of human neurons. Each hidden node receives a combination of input variables and delivers it to the target variable. The neural network model is shown in Figure 2.

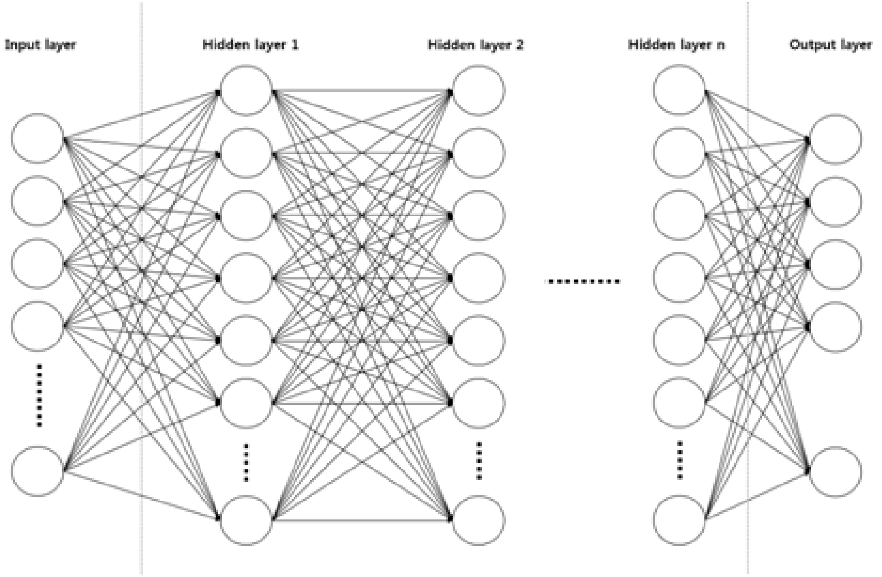

Figure 2. Neural network model

In this case, the coefficients used for combining are called connection strengths. The activation function converts each input value and outputs it to another node to be used as a new input. Among various models, the multi-layer model (MLP) neural network is widely used for data analysis. MLP is an unidirectional neural network composed of hidden layers and output layers composed of input layer hidden nodes. The input layer is composed of nodes corresponding to each input variable. The hidden layer is composed of several hidden nodes. Each hidden node is processed by a nonlinear function of the linear combination of the variable values transmitted from the input layer to the output layer or another hidden layer. It has the role of delivery. The output layer consists of nodes corresponding to the target variable, and the size of the output layer varies depending on the characteristics of the model. Currently, machine learning and neural network engineering are used for such actions as a social prediction, situation recognition, and image processing, and their performance is demonstrated by several studies [15], [16],[17],[18].

2.3 Cellular Automata Model

Several techniques and models have been proposed for modeling the spread of forest fires and fires [19],[20],[21],[22].Since our goal is to confirm the spread of fire per unit time, we constructed a simulated environment in a forest fire diffusion model based on cellular automata (CA) for collecting forest fire pattern data. The current CA model is often used as the base simulation model [23],[24], [25].Additionally, a CA-based simulation model for firefighting similar to the simulation environment of the proposed system has been studied [26], [27],[28],[29].CA also called a cell autologue, is a discrete dynamic system devised by American mathematicians Stanislaw Ulam and John von Neumann in the 1940s that has been studied in computational theory, mathematics, physics, complex systems, mathematical biology, and micro structural models. This model is important because it can effectively model real natural phenomena reflecting diverse dynamic phenomena, such as discrete time. The space of cellular automaton, which is the number of cellular automata, is divided into discrete ‘cells.’ Every cellular automaton belonging to each cell consists of a uniform, regular grid with discontinuous variables. Their state at a particular time is determined solely by the variables of each cell, and the variables are influenced by the values of the neighborhood variables of the previous time step. Neighborhood means a cell adjacent to a cell. All cells change simultaneously based on neighboring variables of previous time according to the local rule. Therefore, according to local interaction and neighboring rules, the CA state transition is identified. CA has a Moore method and a von Neumann method. As shown in Figure 3, the Moore method is a method of performing the state transition for each of the eight adjacent cells of the reference cell every step. Alternatively, there is the von Neumann method that considers only four directions of up, down, left, and right.

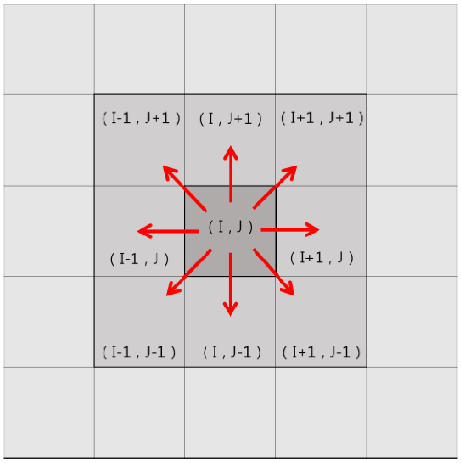

Figure 3.Moore style CA Model

We use Moore's CA and the cells where the state transition occurs in the Moore method are the same as those shown in Figure 3’s(i, j), (i, j + 1), (I, j + 1) (I + 1, j + 1), (i + 1, j), and (i + 1, j + 1) cells. Based on the situation that the learning data cannot be retrieved from the node installed in the actual field, the proposed method constructs the CA-based fire spreading model and obtains the learning data.

3. Proposed Method

3.1. Problem statement

The WSN is installed and used in vast sites for event detection. A false report attack occurring in the application layer is an attack that occurs in order to send a report to the BS containing information about an event that has not occurred, to cause a false alarm or to consume unnecessary energy from the intermediate verification node. Several security protocols are proposed to cover the authenticity of the current report. The existing application layer security protocols depend on verification by the MAC. The security determines whether the report is true or false based on whether the MAC value included in the report is normal or abnormal. This means that if all the MACs included in the report which contains invalid data are normal, all of the reports are regarded as normal events and an inappropriate countermeasure is taken. This is shown in Figure 4.

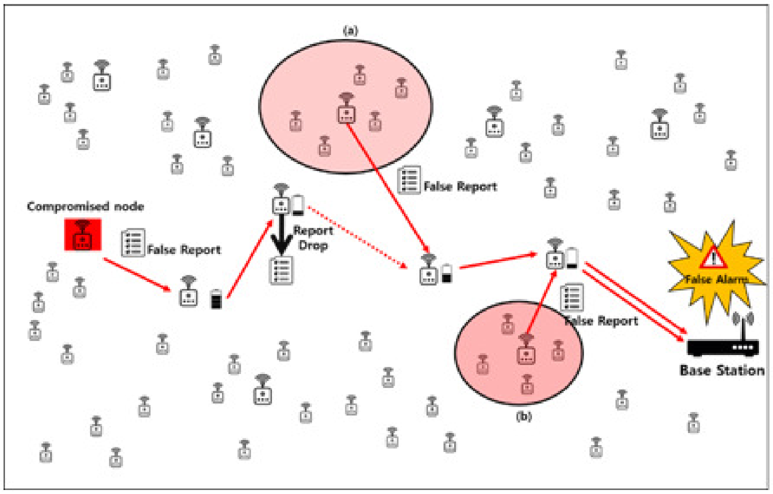

Figure 4. Various attacks in the WSN field

The report generated in the compromised CH is shown in Figure 4 is dropped during the verification process by the security protocol. However, these types of security protocols have lower performance when the attack rate is higher. Figures 4-(a) and (b) are clusters that have major numbers of compromised member nodes. If all the nodes participating in the report are compromised, the attacker can inject a false report attack that includes only the normal MAC. These reports are not verifiable at the validation node and are regarded as normal reports and arrive at the BS securely. The BS finally performs the MAC check of the corresponding report, but all of them are judged as the normal MAC, and a false alarm is sounded. Security protocols that rely solely on MAC can't function in this extreme attack situation. Therefore, a secondary security scheme that does not depend on the MAC is needed. We propose an authenticity discrimination technique of the report contents by recognizing the situation occurring in the field. Although research on tracking the location of corrupted nodes based on context awareness has been performed, it can't provide security against extreme attack situations[30].

3.2. Simulation model

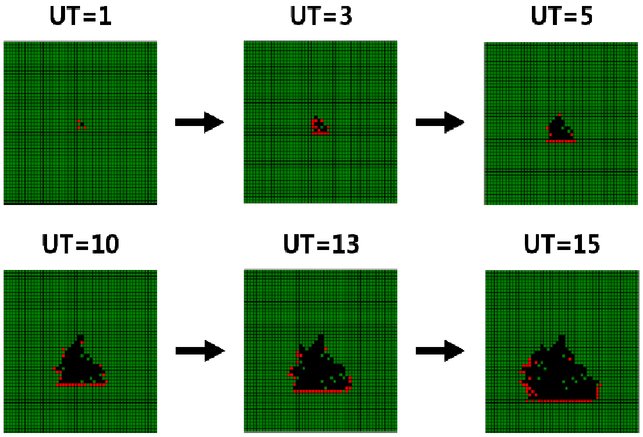

Figure 5. Change of state of the simulation models

The proposed method's simulation model for extracting learning data is based on the Moore method CA. There are three kinds of fire transition states normal, fire in progress, and burnt. Considerations in the forest fire transfer model include the probability of the forest fire transition, wind direction, and wind speed. The probability of fire between two different points is denoted as Pt. The reason for considering the transition rate is that the probability of fire may vary depending on the type of tree and how thick the dry branches or leaves of the tree are on the bottom of the forest. In other words, if there is a place burning in a neighboring grid where the probability of fire is low, it can be a space where the forest fire does not spread. There is no empty space in the simulation space and it depends only on the transition probability. The wind direction changes in 8 directions, the wind speed is 1 - 10, and the fire probability is 50 - 100%. P shows the probability that a tree with a value of Pr is affected by the wind and has a higher probability of fire. Pw is a random variable representing the wind strength [31].The fire transition probability function is as follows:

$$P = P_t + (1 - P_t)P_w.$$

Figure 6 shows a brief overview of the field's fire state transition process when CH = 2500, wind direction = south, wind speed =7, and the maximum unit time (UT) is 15. As shown in Figure 5, fire spreads in the south direction as UT progresses. The cell that is green means a normal state, red is fire in progress, and the burned cell is colored black.

3.3. Overview

The proposed method is described briefly in this section. Each CH periodically sends a report to the BS about the fire situation. If no report is received from the node, the area can be considered a place where no fire occurred. Since the report contains the location information of each node, the BS can collect the reports and draws a real-time situation map for the fire. The relationship between a normal fire transition and the UT is learned through the artificial neural network based on the Map shown in figure.6. We used simulation data using fire models instead of real fire data for neural network engineering experiments because the actual fire data is costly and time-consuming because it must be actually obtained in the field where the node is installed. The CA-based fire prediction model considers wind direction, wind speed, and field conditions. Based on the CA model based data and the evaluation function, the false report second detection algorithm is completed. Figure.6 is a false report second defense scenario of the proposed system.

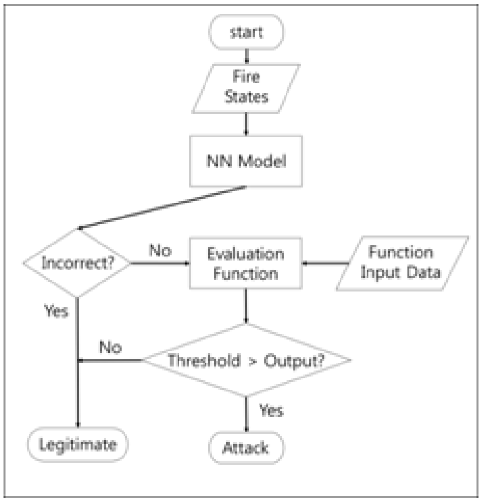

Figure 6. Flow chart of the proposed scheme

The BS starts counting UT when the fire is started. In the BS, the Map generated from reports received from nodes is added to the neural network model for each actual UT to obtain the predicted UT corresponding to the image of the Map. If both UTs are equal, then the BS considers there is no attack occurring at the corresponding UT. If the two values are different, the function argument values are inserted into the evaluation function to determine whether an attack has occurred in the corresponding UT depending on the output of itself.

3.4. Artificial neural network model

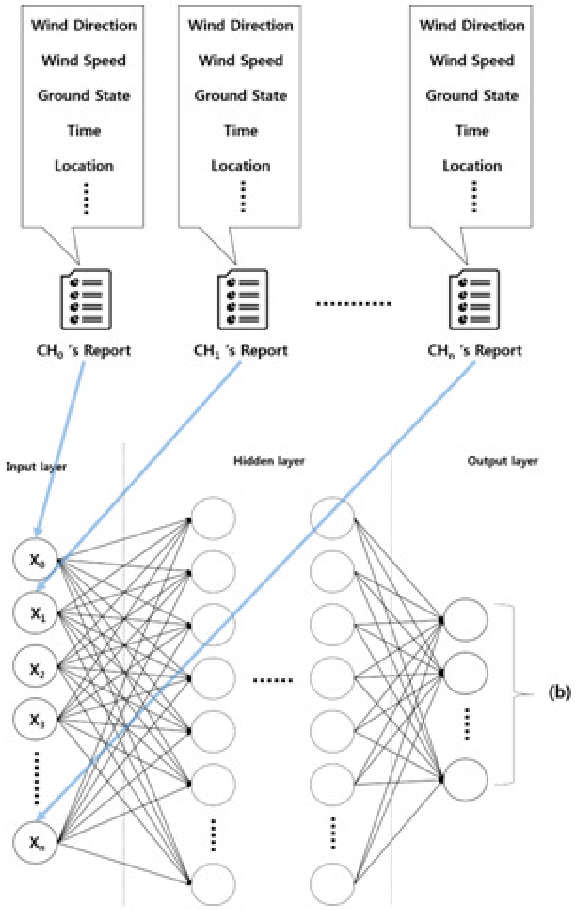

Figure 7. Proposed neural network model.

The input value of the proposed neural network model is the forest fire progress value transmitted from each node. A state can have three values in total: normal, in progress, and burnt. Therefore, the volume of the input layer is the number of CHs installed in the whole field. In the output layer, the UT of the MAP drawn by the corresponding input value is output with a label format. The UT refers to the elapsed time until the state changes at each node. All states go to normal-fire in progress-burnt. The learning rate is 0.01, Relu is considered the activation function, and the dropout is considered to prevent a sticking phenomenon [32-34].

3.5. Evaluation function

The evaluation function of the proposed scheme can be expressed as follows:

$$F = Gap_{UT} * \alpha R_{wp} * \beta R_{ut} * \gamma R_{isp},$$

where Gap_{UT} represents the gap between actual UT and predicted UT. R_{wp}represents reliability of the model depending on the wind power. R_{ut}represents the reliability of the model depending on UT. And R_{isp} represents the reliability of the model depending on the ignition starting position. The minimum gap, which is the evaluation function's threshold value, is 1 because the evaluation function is executed at least when the difference between those values is 1.α, βand γis the weight of each reliability value; the sum of these values is therefore 3. When the result of the function F exceeds 1, the node data coming into the actual UT is regarded as false data that an attack occurred. The reliability of the model, which is the criterion of the evaluation function, depends on how well the model is built. The reliability of the model will be explained in Section 4. The reliability and weight of the model should be able to adjust the scope according to the tendency of the network user.

4. EXPERIMENTAL RESULT

4.1. Experimental environment

Table 1. Experiment parameters

Item	Value
Sensor field size(m*m)	1000 × 1000
Number of sensor nodes	1000 - 36000
Cluster size	40 - 60
Transmission range(m)	80

4.2. Assumptions

It is assumed that all of the attacked reports are false reports of MACs involved in the entire report. Therefore, the BS judges that the report is a normal report. Since the tree firing probability of a simulation is a field's specific characteristics, it does not change after being set initially, which is done at random. The wind direction and wind speed value were also not changed. For the experiment, the only changes made to any values were those that needed to be changed for each graph. All calculations and learning proceed on the BS. We used a tensor flow for machine learning experiments, and the other environmental variables are shown in Table 1. The proposed system is basically used for additional verification on the WSN where the security protocol operates. It is assumed that PVFS is used in the experiment. This protocol is characterized in that the initially set routing path and node position are unchanged. The WSN that the node moves in is a mobile-WSN, which goes beyond the scope of the proposed scheme [35].

4.3. Experimental result

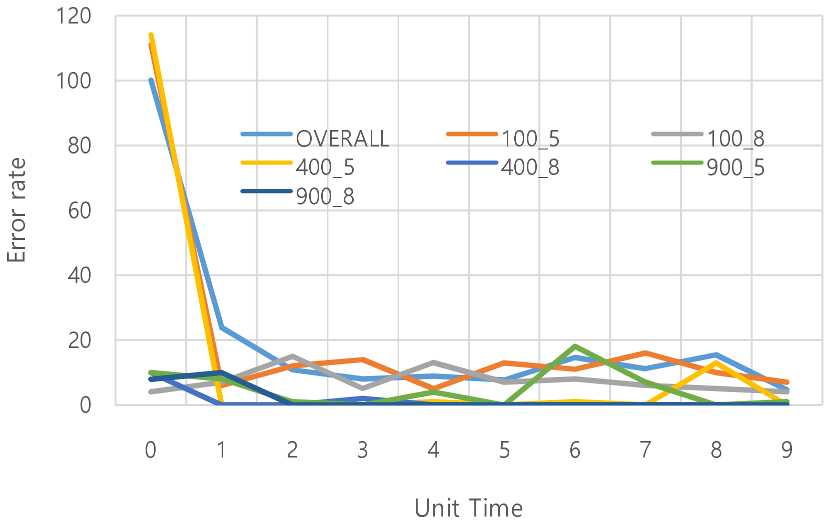

Figure 8. Error rate of neural network model by unit time.

Figure 8 shows the accuracy of the neural network model's prediction performance in each UT when the number of nodes and wind speed are constant. The prediction model has low accuracy at the start of the fire phase and at the end of the fire spread. The reason that UT has a low accuracy between 0 and 1 is that the range of state transitions that one cell can have when it moves one unit of time from the start point is limited. In addition, it is somewhat difficult to accurately discriminate the UT because the fire progresses to the edge of the field and there is no further progression in the field with a small number of nodes. In this particular off-site UT, we can see that most models have fairly high accuracy.

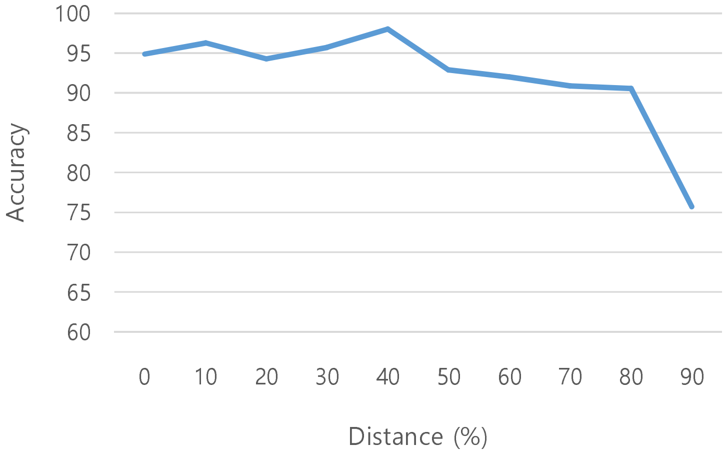

Figure 9. Accuracy of the neural network model by location of the ignition starting point.

The graph shows the prediction accuracy of the neural network model according to the distance from the field center of the fire start point. Let D be the distance from the center of the ignition starting point, i.e., the distance from the center to the end on the same line, as shown in Figure 10. As shown in the graph, a larger D value is more difficult to predict.

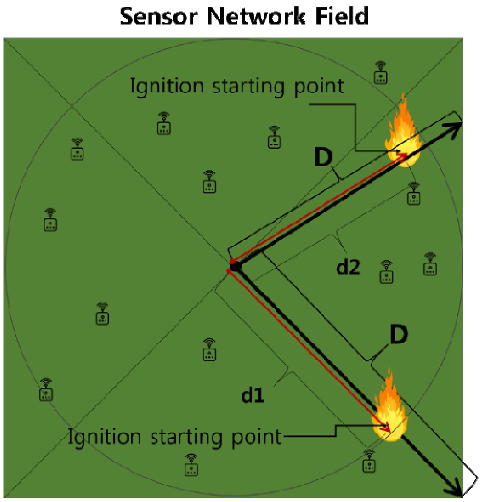

Figure 10. Relative distance of the ignition starting point.

The reason for this is that the neural network model predicts based on the change in the fire state of the MAP per UT. In the case of a fire that starts at the edge of the field or moves to the edge of the field, no further fire state transition occurs or the probability of state transition occurrence becomes low.

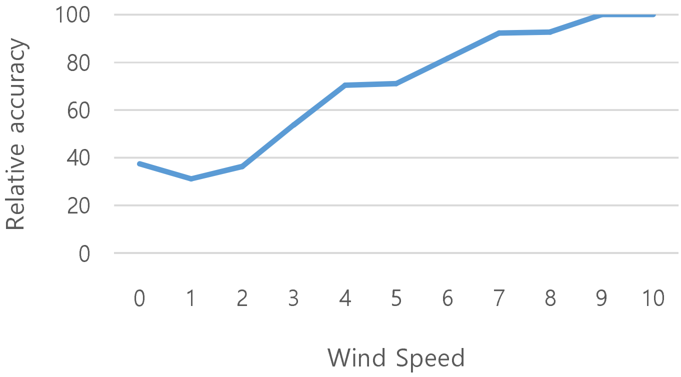

Figure 11. Prediction accuracy of the neural network model by wind velocity.

Figure 9 represents the relative prediction accuracy of the model by the wind speed when the number of nodes is 400 and the unit of time is 10. The neural network model has higher accuracy with high wind speed and the difference in the error rate is approximately twice that of the maximum wind speed and low wind area.

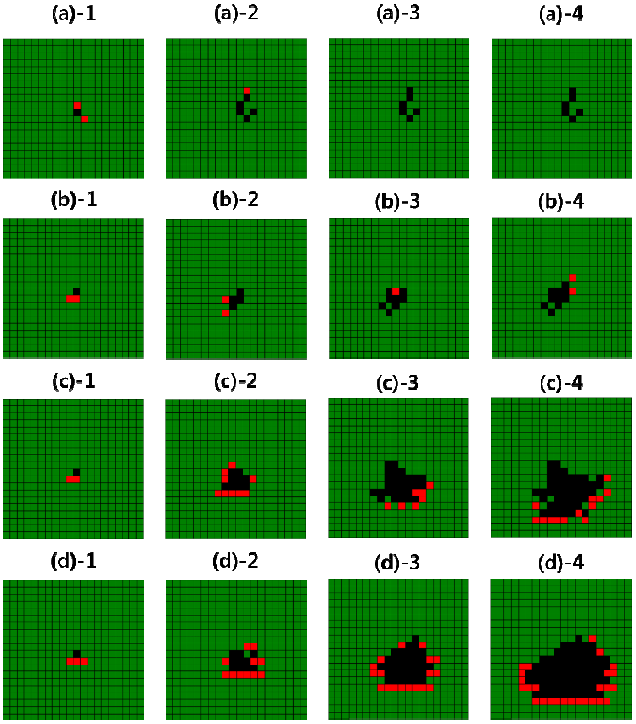

Figure 12. Simulation results according to wind speed.

The reason for the large error failure is shown in Figs. 10 (a) - (d) with the fire state variation when the wind velocity is 0, 3, 5, and 10, respectively. The state transition occurs up to a total of 10 units of time, and the picture shows some of the processes. In the case of the low wind speed, the spread of the fire does not proceed and it automatically evolves at the initial stage of ignition. Therefore, the state transition does not occur even if the unit time progresses and it seems to be a false detection in the model. Considering this special situation, the wind speed is used in the proposal evaluation function.

5. Conclusions

Attacks in WSNs become increasingly diverse and more intelligent and can result in unfiltered reports of false reports due to cluster node deportation. The existing MAC-based authentication process did not provide security in this situation. Context awareness using the machine learning-based prediction model and the evaluation function of the proposed method was used to achieve WSN security. The secondary security scheme of the proposed scheme has the advantage of high scalability. The proposed method can be applied to the whole range of the report without limitation of the model or situation as long as the format of the report shows the status value and unit time. Therefore, we will introduce a model that will apply the proposed method to other models (tank movement, precipitation warning, etc.) in the future. We will also conduct additional tests to determine security accuracy through the evaluation function. Currently, it is only possible to distinguish whether or not an attack has occurred in a unit of time. In future research, we plan to conduct a detection study on a node in which a false report injection attack occurred at the time of an attack.

ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2015R1D1A1A01059484)

REFERENCES

[1] Al-Karaki, Jamal N., and Ahmed E.Kamal. "Routing techniques in wireless sensor networks: a survey." IEEE wireless communications 11.6 (2004): 6-28.

[2] Perrig, Adrian, John Stankovic, and David Wagner. "Security in wireless sensor networks." Communications of the ACM 47.6.

[3] Karlof, Chris, and David Wagner. "Secure routing in wireless sensor networks: Attacks and countermeasures." Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on. IEEE, 2003.

[4] Padmavathi, Dr G., and Mrs Shanmugapriya. "A survey of attacks, security mechanisms and challenges in wireless sensor networks." arXiv preprint arXiv:0909.0576 (2009).

[5] Pathan, Al-Sakib Khan, Hyung-Woo Lee, and Choong Seon Hong. "Security in wireless sensor networks: issues and challenges." Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference. Vol. 2. IEEE, 2006.

[6] Li, Feng, and Jie Wu. "A probabilistic voting-based filtering scheme in wireless sensor networks."Proceedings of the 2006 international conference on Wireless communications and mobile computing. ACM, 2006

[7] Zhu, Sencun, et al. "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks." Security and privacy, 2004. Proceedings. 2004 IEEE symposium on. IEEE, 2004

[8] Yang, Hao, and Songwu Lu. "Commutative cipher based en-route filtering in wireless sensor networks." Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th. Vol. 2. IEEE, 2004.

[9] Yu, Zhen, and Yong Guan. "A dynamic en-route scheme for filtering false data injection in wireless sensor networks." Proceedings of the 3rd international conference on Embedded networked sensor systems. ACM, 2005

[10] Hagan, Martin T., et al. Neural network design. Vol. 20. Boston: Pws Pub., 1996.

[11] Adeli, Hojjat, and Shih-Lin Hung. Machine learning: neural networks, genetic algorithms, and fuzzy systems. John Wiley & Sons, Inc., 1994.

[12] Haykin, Simon S., et al. Neural networks and learning machines. Vol. 3. Upper Saddle River, NJ, USA:: Pearson, 2009.

[13] Weiss, Sholom M., and Casimir A. Kulikowski. Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning, and expert systems. Morgan Kaufmann Publishers Inc., 1991.

[14] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.

[15] Kubat, Miroslav, Robert C. Holte, and Stan Matwin. "Machine learning for the detection of oil spills in satellite radar images." Machine learning 30.2-3 (1998): 195-215.

[16] Sebastiani, Fabrizio. "Machine learning in automated text categorization." ACM computing surveys (CSUR) 34.1 (2002): 1-47.

[17] Bradley, Andrew P. "The use of the area under the ROC curve in the evaluation of machine learning algorithms." Pattern recognition 30.7 (1997): 1145-1159.

[18] Nasrabadi, Nasser M. "Pattern recognition and machine learning." Journal of electronic imaging 16.4 (2007): 049901.

[19] Bak, Per, Kan Chen, and Chao Tang. "A forest-fire model and some thoughts on turbulence." Physics letters A 147.5-6 (1990): 297-300.

[20] Preisler, Haiganoush K., and Alan A. Ager. "Forest-Fire Models." Encyclopedia of environmetrics 3 (2006).

[21] Anderson, D. H., et al. "Modelling the spread of grass fires." The ANZIAM Journal 23.4 (1982): 451-466.

[22] Weber, R. O. "Modelling fire spread through fuel beds." Progress in Energy and Combustion Science 17.1 (1991): 67-82.

[23] Soares-Filho, Britaldo Silveira, Gustavo Coutinho Cerqueira, and Cássio Lopes Pennachin. "DINAMICA—a stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier." Ecological modelling 154.3 (2002): 217-235.

[24] Mallet, Daniel G., and Lisette G. De Pillis. "A cellular automata model of tumor–immune system interactions." Journal of theoretical biology 239.3 (2006): 334-350.

[25] Dijkstra, Jan, Joran Jessurun, and Harry JP Timmermans. "A multi-agent cellular automata model of pedestrian movement." Pedestrian and evacuation dynamics (2001): 173-181.

[26] Karafyllidis, Ioannis, and Adonios Thanailakis. "A model for predicting forest fire spreading using cellular automata." Ecological Modelling 99 (1997): 87-97.

[27] Encinas, A. Hernández, et al. "Simulation of forest fire fronts using cellular automata." Advances in Engineering Software38.6 (2007): 372-378.

[28] Karafyllidis, Ioannis. "Design of a dedicated parallel processor for the prediction of forest fire spreading using cellular automata and genetic algorithms." Engineering Applications of Artificial Intelligence 17.1 (2004): 19-36.

[29] Wolfram, Stephen. "Universality and complexity in cellular automata." Physica D: Nonlinear Phenomena 10.1-2 (1984): 1-35.

[30] Nam, Su Man, and Tae Ho Cho. "Context-aware architecture for probabilistic voting-based filtering scheme in sensor networks." IEEE Transactions on Mobile Computing 16.10 (2017): 2751-2763.

[31] Song and Lee, 2013: Sensitivity Analysis on Ecological Factors Affecting Forest Fire Spreading: Simulation Study Korean Journal of Agricultural and Forest Meteorology, Vol. 15, No. 3 pp. 178~185

[32] Agostinelli, Forest, et al. "Learning activation functions to improve deep neural networks." arXiv preprint arXiv:1412.6830 (2014).

[33] Le, Quoc V., et al. "On optimization methods for deep learning." Proceedings of the 28th International Conference on International Conference on Machine Learning. Omnipress, 2011.

[34] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from over fitting." The Journal of Machine Learning Research 15.1 (2014): 1929-1958.

[35] Munir, Saad Ahmed, et al. "Mobile wireless sensor network: Architecture and enabling technologies for ubiquitous computing." Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on. Vol. 2. IEEE, 2007.

AUTHORS

Sanghyeok Lim Received a B.S. degree in Digital Information Engineering from Hanguk University of Foreign Studies in 2017, and is now working toward an M.S. degree in the Department of Electrical and Computer Engineering at Sungkyunkwan University.

Taeho Cho Received a Ph.D. degree in Electrical and Computer Engineering from the University of Arizona, USA, in 1993, and B.S. and M.S. degrees in Electrical and Computer Engineering from Sungkyunkwan University, Republic of Korea, and the University of Alabama, USA, respectively. He is currently a Professor in the College of Information and Communication Engineering, Sungkyunkwan University, Korea.