Webology (ISSN: 1735-188X)
Volume 18, Number 6, 2021

Cluster Based Malicious Node Detection Method In Wireless Sensor Networks

S. Saminathan[1] , Dr. A.Vinayagam[2]

[1]Research Scholar, PG & Research Department of Computer Science, Government Arts College (Autonomous) (Affiliated to Bharathidasan University, Tiruchirappalli), Karur, Tamilnadu, India.

[2]Assistant Professor, PG & Research Department of Computer Science, Government Arts College (Autonomous) (Affiliated to Bharathidasan University, Tiruchirappalli), Karur, Tamilnadu, India.

ABSTRACT

The development of Wireless Sensor Networks (WSNs) in healthcare domain is picking up force through the expanding cluster of wearable crucial location tags and sign sensors which can track healthcare services and patient location/status persistently in progressive mode. In spite of the expanded scope of potential application systems going from in-hospital, home monitoring, pre-hospital facility, and mobile, to long term database gathering for longitudinal pattern investigation, the security hole between existing WSN plans and the prerequisites of medicinal applications stays uncertain. For the most part, WSN gadgets are amazingly constrained regarding communication, power, and computation. They are frequently deployed to reach areas, in this way expanding security vulnerabilities. The multicast transmission, critical information prioritization, dynamic ad-hoc topology, awareness of location, and coordination of assorted sensors of healthcare applications additionally intensify the security challenges. This paper depicts the cluster based malicious node detection technique for the identification of malicious node in the WSN.

KEYWORDS: Wireless Sensor Network, Trust Calculation, Cluster based Routing, Energy Management, Cluster Formation, Packet Delivery Ratio**.**

1. INTRODUCTION

Wireless sensor networks (WSNs) [1] are made up of small nodes that can sense, compute, and communicate wirelessly. Many routing, data dissemination and power management protocols have been developed expressly for WSNs, where energy efficiency is a key design consideration.

Wireless Sensor Networks provide a significant benefit for a variety of applications in our daily lives. Habitat monitoring, home automation, battlefield surveillance, and intelligent agriculture are just a few examples of real-world applications [2]. In WSN systems, the sensor node detects the data of interest, processes it with the help of an in-built microprocessor, and sends the results to a base station or sink. A wireless sensor network can be created by connecting millions of sensor nodes together. A sensor node is a microprocessor-based embedded device that combines several microprocessor components into a single chip. Despite

the fact that sensor nodes are capable of sensing, data processing, and communication, their limited memory capacity, battery power, bandwidth, and computational power make them vulnerable to several types of attacks [3].

Wireless sensor networks (WSNs) have the potential to monitor huge areas with great temporal and spatial resolution. However, the nodes' small size and inexpensive cost make them appealing for mass deployment, but they come with the drawback of low operational reliability [4]. Mechanical/electrical issues, environmental deterioration, battery depletion, or aggressive manipulation can all cause a node to fail. In fact, due to the often restricted energy budget of nodes powered by small batteries, node failure is projected to be relatively prevalent. Furthermore, if one of the nodes fails, the quantity of multihop routes in the network will be reduced [5]. As a result of such failures, a subset of nodes that have not broken can become isolated from the rest, resulting in a "cut." As a result, if there is no path among two nodes, they are considered to be disconnected.

2. RELATED WORKS

Chatterjee, Pushpita, et al [6] developed a unique distributed safe trust aware clustering mechanism for data delivery that is secure. A trust model is proposed that calculates a node's trust based on recommendation and self-evidence from one-hop neighbors. The suggested clustering protocol divides the network into one-hop disjoint clusters and elects a Cluster head from among the most qualified and trustworthy nodes.

Tan, Shuaishuai, Xiaoping Li, and Qingkuan Dong [7] To address this problem, a trust-based routing technique was developed. A trust reasoning model based on fuzzy Petri net is described in this technique to evaluate trust values of mobile nodes. In addition, a trust-based routing algorithm is presented to select a path with the highest path trust value among all feasible paths to minimize compromised or malicious nodes.

Ishmanov, Farruh, et al [8] In terms of enhancing security and collaborating successfully, trust is critical in wireless sensor networks (WSNs). Traditional security services become more resilient and reliable when trust management (TM) ensures that all interacting nodes are trustworthy during authorisation, authentication, or key management. Furthermore, by assisting in the discovery of dependable nodes, TM promotes node collaboration, which is critical for system performance enhancement.

Gong, Pu, Thomas M. Chen, and Quan Xu [9] introduced Secure and Energy Aware Routing Protocol (ETARP), a new routing protocol aimed for energy efficiency and security in wireless sensor networks (WSNs). ETARP is a project that aims to handle with WSN applications that operate in harsh conditions like the battlefield. The routing protocol's most important feature is route selection based upon utility theory. The idea of utility is an innovative method to including energy efficiency and route dependability into the routing protocol at the same time. When compared to the typical AODV (Ad Hoc On Demand Distance Vector) routing protocol, ETARP discovers and selects routes based on greatest utility while incurring greater overhead costs.

Ahmed, Adnan, et al [10] introduced a Trust and Energy Aware Secure Routing Protocol (TESRP) for Wireless Sensor Networks (WSN) that uses a distributed trust architecture to find and isolate problematic nodes. When determining routing decisions, TESRP uses a multi-facet routing method that considers the trust level, hop counts, and residual energy of surrounding nodes. This technique provides data distribution via trusted nodes while also balancing energy usage between trusted nodes while travelling shorter pathways.

3. PROBLEM STATEMENT

When a node in a Wireless Sensor Network fails, the network is divided into distinct pieces as the topology of the network changes, but the network continues to function. However, the partition has an impact on network dependability, data loss, network QOS, efficiency, and data processing speed. Because if any data is sent in the wrong direction, it will result in data loss, as well as demonstrating the network's instability. Due of its highly restricted environmental degradation and energy budget, node failure is projected to be quite common in WSN. This is most often the case for sensor networks used in difficult and risky areas, such as forest fire monitoring. When a lot of sensors fail for any cause, the network topology may become disconnected, which is then referred to as a set of node failure. The nodes that haven't failed are now cut off from the rest of the network. Node state transitions and node mobility due to the usage of energy efficient or power management strategies can cause network topology changes, which can be identified as node failures. The difficulty of failure management is substantially increased in a highly dynamic network. When a node fails unexpectedly, the network's redundancy and backup pointers are used to re-establish the destroyed links.

4. PROPOSED CLUSTER BASED MALICIOUS NODE DETECTION IN WIRELESS SENSOR NETWORKS

In this contribution, for securing the routing, a reactive on-demand cluster-based Malicious Node identification method for the detection of malicious node has introduced. This approach is extended to develop the communication type of attack in the network. It permits the Malicious node (MN) to interact with a legitimate node directly. So, therefore, to obtain the safe route; initially the proposed method institutes the source for a reliable location by providing a device to discriminate the positive nodes from malicious ones. Then, it establishes the system into 1-hop split clusters, whereas each node selects the positive and best competent node of its 1-hop neighbors to be its cluster-head. The cluster associates in proposed method transmit the packet over the trusted cluster heads (CHs). This proposed process comprised of the subsequent steps to determine the malicious node in the network.

Stage 1: Cluster Formation: This proposed approach executes the weighted clustering algorithm (WCA) [15] to choose CHs procuring into consideration the mobile nodes battery power, mobility, ideal degree (number of neighbors), and transmission power. Unlike WCA, the proposed approach necessitates security into the reckoning to develop trusted clusters. In proposed method every node computes its own weight V_a as follows:

$$V_a = D_a a_1 + L_a a_2 + S_a a_3 + B_a a_4$$

where

D_a represents the difference in the degree among the number of nodes and the immediate neighbors that CH can preferably manage (δ).

L_a depicts the Summation in Distance among a node and its next adjacent. The impulse of La has principally associated with power dissipation. That has recognized that increased energy has needed to interface upon a more significant distance.

S_a represents the node's mobility or speed: A hub by limited versatility is continually a reliable option for a CH.

B_a is the Battery Power of a node. A CH is assumed to use higher battery potential than a normal node because it has more reliability to send out. The weighting should orbitally 1 i.e. $a_1 + a_2 + a_3 + a_4$

CH – Cluster Head
Initially each node A in WSN holds undecided state and opinion of (0, 0, 1) then for each node A

Step 1: Cluster Formation Method

Step 1.1: Compute W_A
Step 1.2: Broadcast W_A to immediate neighbors
Step 1.3: If A receives W_x then
insert W_x in possible CH set
Step 1.4: find node B with minimal weight in possible CH set
Step 1.5: If B = A then
A elects itself as CH
Step 1.6: else if ($a_b^a > a_t$) OR ($a_b^a \leq a_t$ AND $r_b^a \leq r_t$AND $d_b^a \leq d_t$) OR ($d_b^a > d_t$) then
A sends join cluster message to B (If B is a CH or not yet a cluster member, it sends join cluster message to B.
Step 1.7: If A receives Accept Join from B then
A becomes a member of this cluster
Step 1.8: Else
remove B from the possible CH set and Go to line 1.4
Step 1.9: else if $r_b^a > r_t$ then
Remove B from the possible CH set and Go to line 1.4.

Stage 2: Node Trust Calculation: In this method, the gathering of information about the other node by another node gives the trust value calculation. At the different layers of protocol, the appropriate taps are implemented for collecting the forwarded, overhead and received packets. The trust between the two nodes has represented in a 3-dimensional opinion metrics (Acceptance, Rejection and Doubtful)

$t_y^x = (a_y^x , r_y^x, d_t^x)$ such that $a_y^x + r_y^x + d_y^x = 1$

t_y^x indicates the node X's estimation on any node Y's trustworthiness

a_y^x means the faith that X endures for Y

r_y^x indicates the mistrust that X carries for Y (i.e., the likelihood that X cannot trust a node Y).

d_t^x expresses the doubtfulness that X carries for Y (i.e., doubt pervades the void in the inadequacy of both acceptance and rejection).

In the proposed CBMN, concerning the trustworthiness, each node observes another nodes' performance to obtain and document all plus (p) and minus (m) results. As before-mentioned, the metrics of estimation of t_y^x can be represented as a role of p and m as follows:

$$a_y^x = \frac{p}{p + m + 2}$$

$$r_y^x = \frac{m}{p + m + 2}$$

$$d_t^x = \frac{2}{p + m + 2}$$

To compute the sincerity of a hub, every of the acceptance, rejection, and doubtfulness states may differ within 0 and 1 general. At method beginning, every node takes an estimation state of (0,0,1) for every of its next neighbors, with p = m = 0. Each hub controls its next adjacent on a fixed base & registers the amount of minus, & plus results. P value is incremented to 1. Plus, results match towards the contemporaries about strong responses, and positive acknowledgments, appropriate packet forwarding, or some particular situation a consumer would prefer to measure, afforded sufficient mechanisms are accessible. On the next side, minus results raise the state m by 1 and append: denying to transmit packet unless it contains malicious behavior or to preserve energy (selfish nodes), transmitting responses abnormally or route requests, creating an abnormal route responses or request, data altering, otherwise somewhat explicit experience the consumer should choose to evaluate which are accessible. At each point, amount about minus & plus results in variations, the same state of evaluation will be recomputed utilizing the previous equation. In the proposed CBMN, the node's sincerity has determined consorting in the below table 1.

Table 1: Node Trust Calculation in proposed CBMN

Case	When
Case 1: accept a node at the particular time	$> d_t$
Case 2: reject a node	$> r_t$
Case 3: accept a node	$> a_t$
Case 4: for the particular time accept a node	$\leq a_t \leq r_t \leq d_t$

Stage 3: Local Cluster Formation

In the proposed method, a node should shift its CH when its converts into harmful node (i.e. disbelief value > d_t) through requesting a procedure known as the construction of local cluster to dodge the overhead created by reinvading the cluster formation algorithm by every node in the network. In this stage, the node endeavoring to adjust its malicious CH determines the node with the least weight within its 1-hop adjacent. If this node fills the belief circumstances of the above table, that will make to convert the CH evade specific formulation about various CHs

by none branches. It occurs whenever a CH shift to harmful, most significant about its previous branches that 1-hop adjacent is the members of a cluster, would turn CHs.

Step 3: Formation of Local Cluster: Every node keeps the weight of its 1-hop neighbors. Suppose node X is requesting the algorithm

Step 3.1: detect node Y with the least weight in desirable CH set
Step 3.2: if Y=X then
X selects as CH
Step 3.3: else if ($a_y^x > a_t$) OR ($a_y^x \leq a_t$ AND $r_y^x \leq r_t$ AND $d_y^x \leq d_t$) OR ($d_y^x > d_t$)
then
Step 3.3.1: X transmits provoked join cluster information to Y
Step 3.3.2: if (Y is not a cluster member) or (Y is a CH) then
Y transmits X accept join
X converts a part of this cluster
Step 3.3.3: else if Y is a cluster member then
Y switches its status to CH and admits X as its member
Step 3.4: else if $r_y^x > r_t$ then
Exclude Y from the potential CH set and Go to line 3.1.

Stage 4: CH handles the Route Request (RR)

A CH accepts the RR of any node in the system, and suddenly it will compare for the trust state of the node and correlate with the doubtful, rejection, and acceptance value. If a node's rejection value is higher than the specified rejection threshold, the node can be considered malicious.

Step 4: CH Handling Route Request (RREQ)

Step 4.1: CH Z gets an RREQ from node X
Step 4.2: if $r_W^Z > r_t$ then
Step 4.2.1: Z rejects the RREQ packet
Step 4.3: else if Z sustained the same RREQ back by equivalent description field then
Step 4.3.1: Z rejects the RREQ packet.
Step 4.4: else
Step 4.4.1: Z registers its address in the Cluster Address record
Step 4.4.2: if W is its neighbor then
Transmit RREQ to W
Step 4.4.3: else
for each adjacent CHs, CH in Z's CAT do
if CH is already in earlier RREQ's Neighboring CH record then
Jump
Else if CH is in Cluster Address list then
then
Jump
Else
Enter CH entry in Adjacent Gateway Node pair/CH
disseminate RREQ

Stage 5: Handling of Malicious Nodes

In this process, if any determines that some consequent hop while the beginning path packet remains malicious, then that attempts to obtain other trustfulness mediator nodes over the subsequent hop into the root route through examining the cache in the routing table for a route to the endpoint.

Step 5: Dealing of Intermediary Malicious Nodes: Consider node X has to transmit, by node Y, the origin packet obtained from node S. Consider node Z is the following hop to move behind Y allowing to the source node

Step 5.1: if $r_S^X > r_t$ then
Step 5.1.2: X discards the packet
Step 5.2: else if $r_Y^X > r_t$ then
Step 5.2.1: X compares its Two-CAT
Step 5.2.2: if Z is accessible by a node diverse than Y then
X alters the source node
X fixes the E flag in the source route packet
X transmits out the packet to the new following hop
Step 5.2.3: else if X attains an option route in its cache to move W then
X changes the source route.
X adjusts the E flag in the source route packet
X transmits the packet to the new subsequent hop
Step 5.2.4: Else
X re-transmits the packet to S with the F flag set

The destination node is getting a flagged data packet.

Step 5.3: if W gets source route data packet with E flag set then
W transmits complimentary RREP by a revamped route to S

Source node getting a flagged data packet

Step 5.4: If S gets source route data packet with F flag set then
Step 5.4.1: S compares in its cache for an option route to the W
Step 5.4.2: If S gets no route then
S remains for backoff time before re-detecting the route to W
Step 5.4.3: else
S transmits data packet by the new route.

5. SIMULATION RESULT AND DISCUSSION

5.1 Simulation Setup

Table 2 depicts the simulation environment for evaluating the performance of the proposed Cluster based Malicious Node (CBMN) Detection.

Table 2: Simulation Setup

Parameter	Value
Simulation Environment	NS-2
Number of Nodes	100 to 300
Area of Simulation	250m X 250m
Initial Node Energy	20000 joules
Size of the Packet	512 Bytes
Simulation Execution Time	300 Seconds
Node Operating Power	10mW
Percentage of Malicious Node	5% and 15%

5.2 Performance Analysis of the proposed CBMN method with 5% Malicious Node

Table 3 represents the clusters formed by proposed CBMN detection method and existing clustering method at 5% malicious nodes. From the table 3, the proposed CBMN method gives more clusters count than using K-Means and Fuzzy C-Means clustering detection method at 5% malicious nodes.

Table 3: Total Cluster count by the Proposed CBMN method and existing clustering method at 5% malicious node

Number of Nodes	Number of Clusters formed		
	Proposed CBMN method	K-Means	Fuzzy C-Means
100	13	9	10
125	21	17	16
150	28	20	19
175	36	25	23
200	39	29	28
225	43	35	34
250	48	41	40
275	52	47	45
300	61	51	49

Table 4 represents the Average Packet Delivery Ratio by proposed CBMN detection method and existing clustering method at 5% malicious nodes. From the table 4, the proposed CBMN method gives increased packet delivery ratio than using K-Means and Fuzzy C-Means clustering detection method at 5% malicious nodes.

Table 4: Average Packet Delivery Ratio by the Proposed CBMN method, K-Means, and Fuzzy C-Means method at 5% malicious node

Number of Nodes	Average Packet Delivery Ratio		
	Proposed CBMN method	K-Means	Fuzzy C-Means
100	0.9525	0.8864	0.8752

125	0.9424	0.8632	0.8527
150	0.9319	0.8549	0.8341
175	0.9047	0.8321	0.8127
200	0.8936	0.8027	0.7974
225	0.8824	0.7912	0.7706
250	0.8741	0.7739	0.7529
275	0.8627	0.7552	0.7385
300	0.8573	0.7369	0.7141

Table 5 depicts the Energy Consumption (in kWh) by the Proposed CBMN method, K-Means, Fuzzy C-Means clustering detection method at 5% malicious node. From the table 5, it is clear that the proposed CBMN detection method consumes less energy than existing clustering methods like K-Means, and Fuzzy C-Means.

Table 5: Energy Consumption (in kWh) by the Proposed CBMN method, K-Means, and Fuzzy C-Means method at 5% malicious node

Number of Nodes	Energy Consumption (in kWh)		
	Proposed CBMN method	**K-Means**	**Fuzzy C-Means**
100	8.52	15.81	16.32
125	9.43	17.64	18.61
150	10.24	18.37	19.27
175	10.86	19.18	20.12
200	11.24	20.43	21.57
225	11.98	21.3	22.41
250	12.34	22.48	23.26
275	13.28	24.87	24.42
300	15.65	25.28	26.97

5.3 Performance Analysis of the proposed CBMN method with 15% Malicious Node

Table 6 represents the clusters formed by proposed CBMN detection method and existing clustering method at 15% malicious nodes. From the table 6, the proposed CBMN method gives more clusters count than using K-Means and Fuzzy C-Means clustering detection method at 15% malicious nodes.

Table 6: Total Cluster count by the Proposed CBMN method and existing clustering method at 15% malicious node

Number of Nodes	Number of Clusters formed		
	Proposed CBMN method	**K-Means**	**Fuzzy C-Means**
100	14	10	12
125	23	18	17

150	29	21	20
175	38	26	24
200	41	31	30
225	45	37	36
250	50	42	41
275	54	49	47
300	63	52	51

Table 7 represents the Average Packet Delivery Ratio by proposed CBMN detection method and existing clustering method at 15% malicious nodes. From the table 7, the proposed CBMN method gives increased packet delivery ratio than using K-Means and Fuzzy C-Means clustering detection method at 15% malicious nodes.

Table 7: Average Packet Delivery Ratio by the Proposed CBMN method, K-Means, and Fuzzy C-Means method at 15% malicious node

Number of Nodes	Average Packet Delivery Ratio		
	Proposed CBMN method	**K-Means**	**Fuzzy C-Means**
100	0.9434	0.8753	0.8641
125	0.9211	0.8541	0.8436
150	0.9128	0.8338	0.8252
175	0.8936	0.8212	0.8016
200	0.8824	0.7916	0.7863
225	0.8613	0.7723	0.7614
250	0.8552	0.7628	0.7428
275	0.8418	0.7441	0.7274
300	0.8361	0.7187	0.7032

Table 8 depicts the Energy Consumption (in kWh) by the Proposed CBMN method, K-Means, Fuzzy C-Means clustering detection method at 15% malicious node. From the table 8, it is clear that the proposed CBMN detection method consumes less energy than existing clustering methods like K-Means, and Fuzzy C-Means.

Table 8: Energy Consumption (in kWh) by the Proposed CBMN method, K-Means, and Fuzzy C-Means method at 15% malicious node

Number of Nodes	Energy Consumption (in kWh)		
	Proposed CBMN method	**K-Means**	**Fuzzy C-Means**
100	10.61	17.91	18.48
125	11.39	18.53	19.53
150	12.33	19.46	20.73
175	13.95	20.27	21.38

200	14.45	22.34	23.66
225	15.76	23.22	24.32
250	16.45	24.59	25.37
275	17.39	25.68	26.53
300	18.74	26.37	27.89

6. CONCLUSION

A novel cluster-based malicious node detection method has been suggested in this work for transmitting packets in a large network and detecting malicious nodes. The trust value computation approach was used in this methodology to determine the trustworthiness of the network's surrounding nodes. The infected node has been removed from the network. When the number of nodes, as well as the number of malicious nodes, present in the network, the packet delivery ratio and detection rates increase.

REFERENCES

[1] Zhang, Tong, Lisha Yan, and Yuan Yang. "Trust evaluation method for clustered wireless sensor networks based on cloud model." Wireless Networks 24.3 (2018): 777-797.

[2] Khan, Zeeshan Ali, and Peter Herrmann. "A trust based distributed intrusion detection mechanism for internet of things." Advanced Information Networking and Applications (AINA), 2017 IEEE 31st International Conference on. IEEE, 2017.

[3] Gupta, Govind P., and Sonu Jha. "Integrated clustering and routing protocol for wireless sensor networks using Cuckoo and Harmony Search based metaheuristic techniques." Engineering Applications of Artificial Intelligence 68 (2018): 101-109.

[4] Gaber, Tarek, et al. "Trust-based secure clustering in WSN-based intelligent transportation systems." Computer Networks146 (2018): 151-158.

[5] Meng, Weizhi, et al. "Enhancing Trust Management for Wireless Intrusion Detection via Traffic Sampling in the Era of Big Data." Ieee Access 6 (2018): 7234-7243.

[6] Chatterjee, Pushpita, et al. "A trust enhanced secure clustering framework for wireless ad hoc networks." Wireless networks 20.7 (2014): 1669-1684.

[7] Tan, Shuaishuai, Xiaoping Li, and Qingkuan Dong. "Trust based routing mechanism for securing OSLR-based MANET." Ad Hoc Networks 30 (2015): 84-98.

[8] Ishmanov, Farruh, et al. "Trust management system in wireless sensor networks: design considerations and research challenges." Transactions on Emerging Telecommunications Technologies 26.2 (2015): 107-130.

[9] Gong, Pu, Thomas M. Chen, and Quan Xu. "ETARP: An energy efficient trust-aware routing protocol for wireless sensor networks." Journal of Sensors 2015 (2015).

[10] Ahmed, Adnan, et al. "A secure routing protocol with trust and energy awareness for wireless sensor network." Mobile Networks and Applications 21.2 (2016): 272-285.

[19] Subhashini, M., & Gopinath, R., Mapreduce Methodology for Elliptical Curve Discrete Logarithmic Problems – Securing Telecom Networks, International Journal of Electrical Engineering and Technology, 11(9), 261-273 (2020).

[20] Upendran, V., & Gopinath, R., Feature Selection based on Multicriteria Decision Making for Intrusion Detection System, International Journal of Electrical Engineering and Technology, 11(5), 217-226 (2020).

[21] Upendran, V., & Gopinath, R., Optimization based Classification Technique for Intrusion Detection System, International Journal of Advanced Research in Engineering and Technology, 11(9), 1255-1262 (2020).

[22] Subhashini, M., & Gopinath, R., Employee Attrition Prediction in Industry using Machine Learning Techniques, International Journal of Advanced Research in Engineering and Technology, 11(12), 3329-3341 (2020).

[23] Rethinavalli, S., & Gopinath, R., Classification Approach based Sybil Node Detection in Mobile Ad Hoc Networks, International Journal of Advanced Research in Engineering and Technology, 11(12), 3348-3356 (2020).

[24] Rethinavalli, S., & Gopinath, R., Botnet Attack Detection in Internet of Things using Optimization Techniques, International Journal of Electrical Engineering and Technology, 11(10), 412-420 (2020).

[25] Priyadharshini, D., Gopinath, R., & Poornappriya, T.S., A fuzzy MCDM approach for measuring the business impact of employee selection, International Journal of Management, 11(7), 1769-1775 (2020).

[26] Poornappriya, T.S., & Gopinath, R., Application of Machine Learning Techniques for Improving Learning Disabilities, International Journal of Electrical Engineering and Technology, 11(10), 403-411 (2020).

[27] Poornappriya, T.S., & Gopinath, R., Rice Plant Disease Identification using Artificial Intelligence Approaches, International Journal of Electrical Engineering and Technology, 11(10), 392-402 (2020).