Received: 14 October 2019 | Revised: 25 January 2020 | Accepted: 23 February 2020
DOI: 10.1002/dac.4394

RESEARCH ARTICLE

WILEY

ORuML: Optimized Routing in wireless networks using Machine Learning

Sapna Chaudhary | **Rahul Johari**

SWINGER: Security, Wireless, IoT Network Group of Engineering and Research Lab, University School of Information, Communication and Technology (USIC&T), Guru Gobind Singh Indraprastha University, Sector-16C, Dwarka, Delhi, India

Correspondence
Rahul Johari, USIC&T, GGSIPU, Delhi, New Delhi, 110078, India.
Email: rahul@ipu.ac.in

Summary

Routing is a process of selecting a path in a network for delivering a packet from source node to destination node. Successful delivery of a message is a challenge, and therefore, this paper proposes an algorithm for a wireless network called Optimized Routing in wireless networks using Machine Learning (ORuML), which uses machine learning algorithm namely, K-nearest neighbor (KNN), Support Vector Machine (SVM), and Multinomial Logistic Regression (MLR), to predict the network type of the source and destination nodes. The ML model is trained by using characteristic features of a node collected in real time such as battery power utilization, available internal storage, IP address, and range of a node. Intuitively, the MLR should outperform KNN and SVM in terms of accuracy and Area under ROC Curve (AUC). The proposed algorithm determines whether the source and destination nodes are co-located and also, determines the best neighboring hop for efficient routing.

KEYWORDS

accuracy, machine learning, multinomial logistic regression, wireless network, optimized routing, K-nearest neighbor, support vector machine

1 | INTRODUCTION

Wireless network is a network of electronic devices where the communication of data between the devices takes place over the air. It contains Bluetooth, Mobile Ad-hoc Network (MANET), Delay Tolerant Network (DTN), and many other networks. In a Bluetooth network, there are no intermediate nodes as the message can be transferred from master node to slave node directly, whereas in MANET and DTN, the intermediate nodes are needed for successful communication between the source and destination node. The successful delivery of a message to its destination node highly depends on the intermediate nodes. In this paper, an algorithm ORuML has been proposed for efficient delivery of a message by considering node characteristic features collected in a real time. ORuML uses machine learning techniques to train a model on various features of a node such as IP address, MAC address, available internal storage, range of a node, and battery power utilization of a node. The model predicts the type of a network to which the source and destination nodes belong to, and for MANET and DTN, the best hop selection takes place.

Notations: S_N, Source Node; D_N, Destination Node; C_n, Class of a node; B_n, Battery power utilization of a node; I_n, Internal storage of a node; G, Graph; E, number of edges in a graph; G_e, Graph edge; R_n, Random number; E_g, set of edges in a graph in pair of from node and to node; F_n, From node; T_n, To node; N_{id}, Network id; N_t, Network type; SN, Same Network; DN, Different Network; H_o, Homogeneous network; H_t, Heterogeneous network; S_n, Set of nodes in a graph; T_f, Trust factor; N, Number of nodes in a network

 © 2020 John Wiley & Sons, Ltd.

For the sake of simplicity and clarity, the rest of the paper is organized as follows. In Section 2, the problem statement of the proposed work has been discussed. In Section 3, the motivation for this study is described. In Section 4, the related work on networking with machine learning is presented. In Section 5, the proposed work is explained in detail. In Section 6, the method used for this work and the algorithms are discussed and explained. In Section 7, the experimental setup of the proposed work is described. In Section 8, the results of the work are presented followed by conclusion and future work.

2 | PROBLEM STATEMENT

When the data are transferred from a source node to a destination node, there are various intermediate nodes that help in transfer of messages. Therefore, the selection of such nodes is necessary for successful and on time delivery of data.

A node with high battery backup and sufficient internal storage can hold a message for long duration of time and is therefore a better candidate for next hop selection. The real time node characteristic features can be used for prediction, and machine learning technique can be applied to predict the class of a node. For example, if the class of a destination node is Bluetooth, then there is a high probability that the message will be delivered successfully, and there is no need to find out the best neighboring node. If the class of destination node is MANET, then the best neighboring node will be selected and same holds true for the Delay Tolerant Network (DTN). Also, when the source and destination nodes are part of same network, the nodes are said to be co-located, and hence, message delivery probability increases.

3 | MOTIVATION

Nowadays, machine learning is touching every field of day-to-day life. It learns from the past experiences, recognizes patterns, and predicts for future based on the learned pattern. It also helps to automate the complex processes and is gaining momentum from the last few years.

To the best of our knowledge, none of the academic researchers has applied the machine learning on network node features, and therefore, the motivation for the proposed work is to apply the machine learning techniques on the node features to classify the network into Bluetooth, MANET, and DTN and then to apply the proposed routing algorithm ORuML as discussed in Section 5. Also, the dataset of network node features has not been used for routing, and therefore, the proposed work combines the routing with machine learning using the features of a network nodes.

There are various classification techniques in supervised machine learning, and the three most widely used algorithms are applied to classify the nodes into three network types in this study so that routing scheme can be applied only on MANET and DTN for the selection of best neighboring node. The classification techniques applied on the dataset are KNN, MLR, and SVM because these are the classification techniques that can be applied on linearly separable data and are robust to noisy data. They are easy to implement, and the training is fast in KNN algorithm.[1] Also, the MLR algorithm can be applied if the dependent variable has more than two nominal categories like Bluetooth, MANET, and DTN network class in network node characteristic features (NNCF) dataset.

The delivery of a message is very important in areas like disaster prone areas where when some disaster occurs, the affected people get help immediately and as per their needs without any delay by delivering a message via optimized path. The various applications where the proposed work can be used are shown in Table 1 and therefore motivates to propose the algorithm.

4 | LITERATURE SURVEY

Turing[2] has given the idea that machine could also be made like a human being. Further, many researchers have adopted his proposal, and thereby, the concept of machine learning came into existence. Till date, machine learning has been used in various real life applications even the field of networking is also not untouched from the influence of machine learning. From time to time, several scientists have tried to optimize the networking activities through machine learning approaches.In Russell et al.,[3] the new routing protocol named wireless adaptive routing protocol (Wrap-5) has been proposed for better decisions of routing in an heterogeneous network. The performance of Wrap-5 has been compared with reinforcement learning algorithm and shows better result in shortest path routing. Ghouti[4] has developed a neural learning-based solution to a problem that occurs due to the advancement in the wireless network and mobility nature of a MANET but in the proposed solution future mobility of a MANET node that may cause change in topology can be efficiently predicted. The proposed predictor uses the traces of real-world mobility and achieves higher accuracy than the existing prediction algorithm as it captures the mobility patterns and interaction between the nodes more accurately. Jiang et al.[5] have studied the application of several machine learning algorithms in perspective of next-generation networks like device-to-device (D2D) networks, heterogeneous networks, massive MIMOs, cognitive radio, and smart grid. The main aim of their paper was to study the role of supervised learning, unsupervised learning, and reinforcement learning algorithms in the context of future networks. Sharma et al.[6] emphasized on the successful delivery of a message in an opportunistic network by proposing a new routing protocol called MLProph in which machine learning techniques, namely, decision tree and neural networks were used for determining the successful message delivery. The predictability value derived from the PROPHET routing scheme, power consumption of a node, a node successful deliveries, a node location, and its speed was used to train the machine learning model.

Ghaffari[7] have proposed an optimal method in which the local data of the neighboring node were used, and the reinforcement learning was used on trial and error basis for choosing the best alternative out of all neighbors of the node for packet transmission. The reinforcement learning was used for predicting the behavioral pattern of a node with respect to the target node. Therefore, the optimal method was proposed by reducing the packet transmission delay and by predicting the patterns of node. The proposed algorithm uses the local information and was designed for routing in MANET. Wang et al.[8] have focused on workflow of machine learning for networking (MLN) that explains method to apply machine learning in networking area. The broad guideline of research in the field of networking with machine learning has been discussed in this paper. The recent advances are reviewed, for example, traffic prediction, traffic classification, resource management, scheduling, and network performance prediction. The various issues are discussed on which the machine learning and networking can together solve the problem and future research can be explored. Mao et al.[9] have proposed a competent and efficient routing table construction technique by using deep learning methodology for a graphical processing unit (GPU)-accelerated software-defined routers (SDRs). They have used the supervised deep belief architecture in order to calculate the next nodes by using traffic patterns of the edge routers. They have also evaluated the time cost and complexity of proposed routing strategy. The authors have found that the proposed routing algorithm runs 100 times more rapidly on GPU than a central processing unit (CPU). Rath et al.[10] have focused on the energy-saving algorithm for a mobile ad hoc network. Conventional algorithms do not care about the energy in nodes while selecting a path for routing and therefore imbalance the level of energy in a network. The nodes with less energy drain off resulting in broken path, and therefore, selected path fails to deliver a message. In order to maintain the reliablity of a system, authors have proposed an algorithm that selects the best energy-saving path for message delivery. Roy et al.[11] have focused on the dumb nodes present in the environment that can detect the surrounding but cannot deliver the sensed data to others nodes in a network. Since the dumb nodes are dynamic, their isolation problem cannot be solved same as the traditional method. The authors have proposed a D3 algorithm that uses cumulative dumb test for detecting dumb nodes behavior in an environment. Li and He[12] have introduced WEBee, an emulation technique for achieving the high throughput of cross-technology communication (CTC) close to the throughput of ZigBee standard.

Musumeci et al.[13] focused on the machine learning applications in optical networking and communication. The authors provide machine learning methods introductory tutorial in the field of optical networks and survey the optical related existing work. The ML methods used in optical networks are discussed, namely, supervised learning, unsupervised learn-

TABLE 1 Application areas of ORuML

Application area	Description
Natural calamity or Disaster prone areas	The ORuML can be used in areas which are more prone to natural calamities so that relief and rescue efforts can reach to affected people quickly.
Dessert area	The areas which are vast and cover a large area and delivery of a message is itself a task, ORuML can be used in such areas.
Forest area	Again, forests cover a large area where delivery of message is not that easy, and therefore, we can use this ORuML in forest areas as well.
Ocean seabed	In order to preserve rich acquatic flora and fauna, it is required to track the activities of fishes and then delivering all the details to the control room responsible for collecting the information, ORuML can be used.

ing, semisupervised learning, and reinforcement learning. Also, they provide comprehension of new research directions. Ayoubi et al.[14] have found the application of machine learning in the area of autonomic network management. They have shown the use of machine learning algorithms to realize the autonomicity in the various areas of network like configuration, fault, security, accounting, and performance. The authors have also illustrated the challenges and opportunities of machine learning for the management of autonomic networks. In Wang et al.,[15] DopplerFi has been proposed that enables the two-way CTC between Bluetooth low energy (BLE) and Wi-Fi by minute shifting of carrier frequency of a sender and thus achieves minimal disturbance in a legacy network.

Challita et al.[16] have shown their interest on the security and wireless challenges that emerge in the unmanned aerial vehicles (UAVs) delivery system, transportation intelligent system, and in multimedia streaming of real time, and therefore, artificial neural network (ANN) solution has been proposed to address such challenges. The ANN approach enables the UAVs to effectively utilize the resources of wireless system and also ensure the secure operation in a real time. Anwar et al.[17] have proposed a machine learning framework to detect and to classify the amateur drones out of the noisy environment. The support vector machine (SVM) was applied on the features extracted by using two techniques named mel frequency cepstral coefficients (MFCCs) and linear predictive cepstral coefficients (LPCCs) along with the help of kernels used for better and accurate classification of sounds and thus has 17% accuracy in detection when compared with detection by correlation-based drone sound. Chaudhary et al.[18] have focused on the concept and applications of internet of things (IoT). Their study states that the message should be delivered immediately in IoT enabled or wireless networks; so to achieve this objective, nodes should be in transmission range of each other. Till date, very few researches have been done for the optimization of IoT enabled or wireless network.

So in this study, routing using machine learning algorithm has been proposed for the wireless networks, namely, Bluetooth, MANET, and DTN. The proposed work is different from the above work as the machine learning technique has been applied on the features of a node to classify the network type to which the node belongs, and if the network is MANET or DTN, then the proposed routing scheme can be applied.

5 | PROPOSED METHODOLOGY

In the proposed work, initially the information about the characteristic features of a network node is collected from individuals by a Google Form,[19] which includes the parameters of a mobile handset like its RAM, CPU, IP address, MAC address, battery power utilization of the phone at that instant of time, and the available internal storage of the phone. The dataset thus created contains the node information present at different location. The dataset collected contained some irrelevant data and therefore, cleaning was done on collected data manually to get the appropriate dataset. The node parameters were used for identification of the best neighboring node for the purpose of routing in a network. The network ID of each node was obtained from its IP address. The best neighboring node was selected out of all the neighboring nodes for delivering a message in a network. If the source node and the destination node network IDs are same, then both the nodes are part of same network, and since the two nodes are co-located, they may be socially inclined and share the common interest with each other.

The best neighboring node was selected on the basis of a class to which it belongs. The class of a node is decided on the basis of following parameters as detailed in the Table 2 and in Section 5.1 .

The best hop node was selected on the basis of the battery power utilization of the phone; if the battery percentage is less at that instant of time, then the probability of delivering a message by that node decreases as the battery of that node may drain off at any time before delivering a message to the destination node. The message cannot be delivered to a node whose available internal storage is less when compared to the size of a message.

For example, the source node N1 wants to send data to the destination node N5, the next hop from node N1 is N2 and the size of a message generated by N1 is 1.5 MB, but the internal storage of the node N2 is 1 MB. Therefore, the node N2 is not the best hop node for delivering a message to the destination node, whereas the node N3 has internal storage of 4 MB, and hence, node N1 will forward a message to node N3. The above example is shown in Figure 1.

Figure 2 shows an example of earthquake in nepal on April 25, 2015. The proposed algorithm can play a major role in such areas. People who got affected due to earthquake as well as people who are volunteering them both need help and delivery of their messages to the remote data center are utmost important. Since the area is disrupted due to earthquake, the delay in messages may be possible thus, it is a delay tolerant network. The remote data center is located in some safe place, and it belongs to a MANET, but here the delivery of message by using an optimized path is very important, and now

TABLE 2 Class distribution based on battery power utilization and internal storage

Network class	Condition
CLASS 1	Battery power utilization ≥ 80% and internal storage ≥ 60 GB OR Battery power utilization ≥ 80% and 60 GB > internal storage > 20 GB OR 80% > battery power utilization > 10% and internal storage ≥ 60 GB OR 80% > battery power utilization > 10% and 60 GB > internal storage > 20 GB
CLASS 2	Battery power utilization ≥ 80% and internal storage ≤ 20 GB OR 80% > battery power utilization > 10% and internal storage ≤ 20 GB
CLASS 3	Battery power utilization ≤ 10% and internal storage ≥ 60 GB OR Battery power utilization ≤ 10% and 60 GB > internal storage > 20 GB
CLASS 4	Battery power utilization < 10% and internal storage < 20 GB

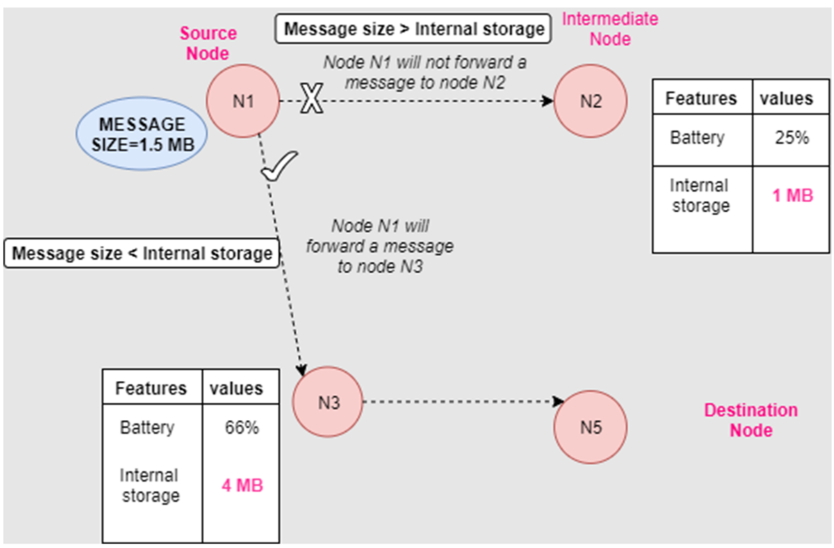

FIGURE 1 Best node selection example

the proposed algorithms come into a picture. By using a machine learning technique, the network type can be predicted of the affected area, and if it belongs to MANET or DTN, the proposed algorithm can be used.

A temporary network has been designed on the fly basis between the mobile phones of all the person, the one who got affected, and the one who are providing relief and rescue operation. For example, if a person requires blanket prior to food that message can be delivered to the remote data center as soon as possible. Also, there may be case that person got stuck inside the building and is badly injured thus cannot deliver a message as he/she may not be in such a condition; in that case, drones will capture the image and send the image to remote data center. By using the proposed algorithm, the best optimized path will be selected, and the probability of message delivery will increase.

In Figure 2 A,B and C are the drones to capture the image and to send and receive a message. There are mobile nodes present in the temporary network, namely, N1, N2, N3, N4, N5, N6, N7, and N8 . The remote data center is located somewhere outside the affected area but close to it. The exchange of messages takes place to deliver a message to remote center for helping people who need help.

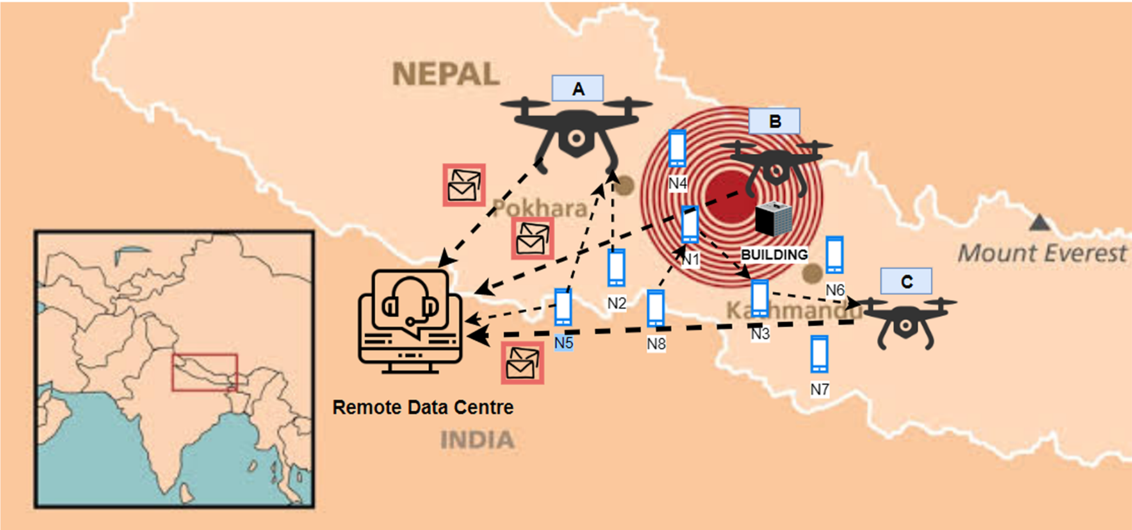

FIGURE 2 Real application scenario of the proposed algorithm[20]

TABLE 3 Distribution of network based on the range of a node

Network class	Condition
Bluetooth	Range ≤10 m
MANET	Range >10 m and range ≤100 m
DTN	Range <100 m

Another vital feature of network node selection is trust factor. The trust factor is calculated between the source node and the destination node/intermediate node, if the source node has communicated with the destination node in the past and has successfully delivered the message, then its trust factor is increased by 1. If in future the message delivery is unsuccessful, then the trust factor between the source and destination node is decreased by 1. Therefore, the hop node having the highest trust factor is considered to be one of the best hop node. The priority is assigned to each message generated by a node range from 1 to number of messages generated by a node. In case of congestion, the message is dropped by the node if the priority of a message is less than the priority of each message generated by the node. The parameter named range is added in the dataset randomly that varies from 1 to 150 for each node present in the dataset, and the type of network is decided on the basis of following parameters as detailed in the Table 3.

If the source node and the destination node are part of same network, then the nodes belong to the homogeneous network otherwise, the two nodes belong to the heterogeneous network. The flowchart of the proposed work is shown in Figure 3. The machine learning was applied on the collected dataset where attribute named network class was selected as target variable and all other attributes of a node as independent variables.

5.1 | Categorization of classes

The node features named battery power utilization (BPU) and internal storage (IS) are used to classify the nodes into different classes. Since the two features are used, so $2^2 = 4$ classes are possible. Each class has three possible cases low, medium and high values of respective feature and therefore in total, there are $3^2 = 9$ possible combinations for the BPU and IS as shown in Figure 4.

In Table 4 for CLASS 1, the value of BPU and IS is 1 that means the battery power utilization and the internal storage of a node are not less and can be considered for message delivery. Similarly for CLASS 2, the BPU value 1 and IS value 0 say that the battery power utilization is good enough, but the internal storage is less than or equal to minimum value, that is 20 GB, and similarly for CLASS 3, the value of BPU value 0 and IS value 1 shows that BPU is not good enough to deliver a message. In CLASS 4, the battery power utilization and internal storage are minimal for delivering a message, and it is the worst node for message delivery.

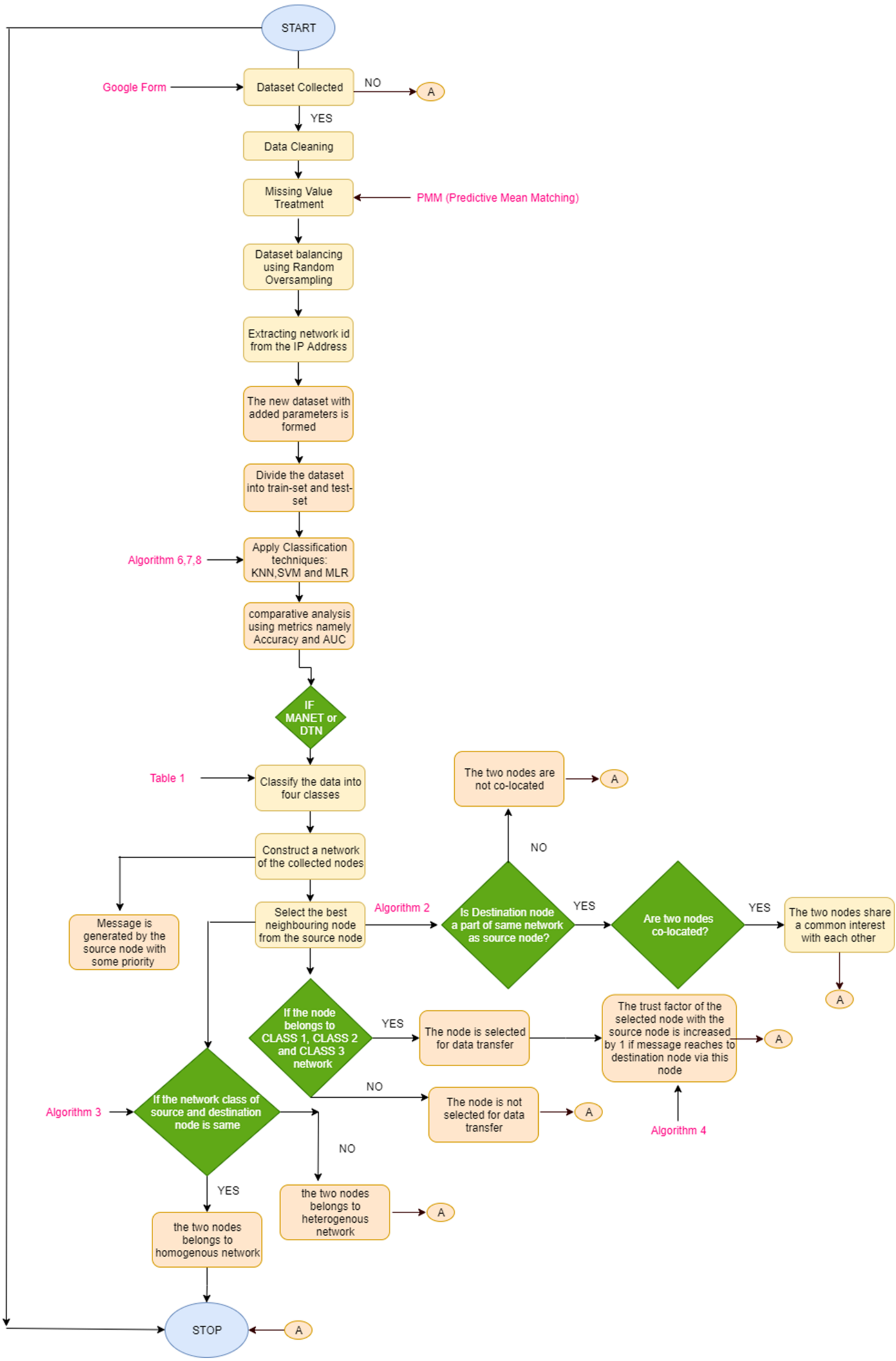

FIGURE 3 Flowchart of working of ORuML algorithm

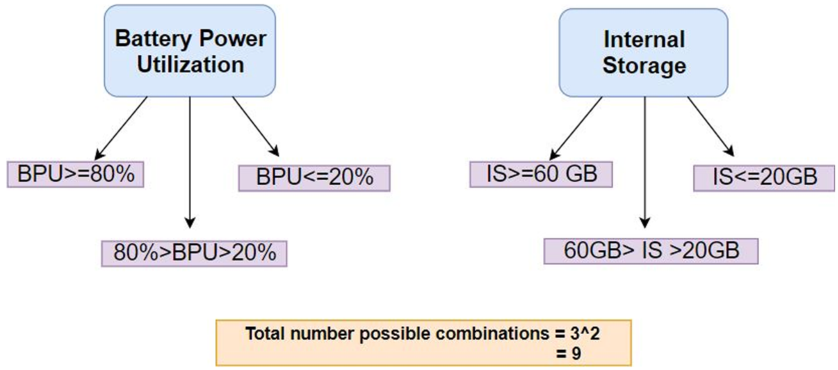

FIGURE 4 Possible combinations of BPU and IS

TABLE 4 Truth table for class of a network node

Battery power utilization	Internal storage	Output class
1	1	CLASS 1
1	0	CLASS 2
0	1	CLASS 3
0	0	CLASS 4

Node_id	RAM	CPU	Battery	INTERNAL	MAC	IP	network_id	network_class	class	RANGE	network_type
1	4	1	100	4	ac:c1:ee:b0:e8:ad	192.168.0.100	192.168.0.0	NETWORK E	CLASS 4	20	MANET
2	4	1.8	8	64	70:bb:e9:74:e5:f6	192.168.31.229	192.168.31.0	NETWORK K	CLASS 4	59	MANET
3	2	1.3	8	16	02:00:00:00:00:00	192.168.2.7	192.168.2.0	NETWORK I	CLASS 4	41	MANET
4	4	2.2	6	64	60:2E:02:C4:3E:E2	10.226.30.164	10.0.0.0	NETWORK A	CLASS 4	62	MANET
5	6	2.45	97	64	94:65:2d:eb:89:53	192.168.0.39	192.168.0.0	NETWORK E	CLASS 2	37	MANET
6	4	2.2	11	64	60:8E:08:43:38:C2	10.226.30.169	10.0.0.0	NETWORK A	CLASS 4	23	MANET
7	4	2	100	64	d8:6c:02:b3:5c:0b	192.168.33.4	192.168.33.0	NETWORK L	CLASS 2	11	MANET
8	4	2	99	32	00:ec:0a:2b:4a:2b	192.168.0.184	192.168.0.0	NETWORK E	CLASS 3	30	MANET
9	1	1.3	93	8	60:2E:02:C4:3E:E2	192.168.2.112	192.168.2.0	NETWORK I	CLASS 4	105	DTN
10	3	1.6	97	64	60:8E:08:43:38:C2	100.86.247.148	100.0.0.0	NETWORK B	CLASS 2	14	MANET
11	3	1.8	97	32	38:A4:ed:5d:f9:13	192.168.31.196	192.168.31.0	NETWORK K	CLASS 3	5	Bluetooth
12	4	1	60	10	e0:62:67:1b:d9:79	123.34.4.16	123.0.0.0	NETWORK D	CLASS 3	48	MANET
13	6	1.6	63	64	54:B8:02:15:34:61	100.65.219.215	100.0.0.0	NETWORK B	CLASS 2	8	Bluetooth
14	8	2.8	34	256	64:a2:f9:c4:2c:18	192.168.5.4	192.168.5.0	NETWORK M	CLASS 1	23	MANET
15	4	1.5	64	16	BC:D1:1F:37:18:C4	56.68.112.45	56.0.0.0	NETWORK Q	CLASS 3	124	DTN
16	4	2	7	2.09	d8:32:e3:e8:8d:54	10.23.254.51	10.0.0.0	NETWORK A	CLASS 4	26	MANET
17	4	2.9	16	64	08:25:25:40:6a:bc	10.226.30.161	10.0.0.0	NETWORK A	CLASS 4	84	MANET
18	8	2.2	50	32	bc:9F:EF:E9:4F:2E	103.92.41.37	103.0.0.0	NETWORK C	CLASS 3	104	DTN
19	4	1.8	97	64	4c:49:e3:67:cc:11	10.88.16.112	10.0.0.0	NETWORK A	CLASS 2	20	MANET
20	4	3.4	97	64	08:25:25:40:6a:bc	10.88.16.117	10.0.0.0	NETWORK A	CLASS 2	56	MANET

FIGURE 5 Screenshot of network node characteristic features (NNCFs) dataset

6 | EXPERIMENTAL SETUP

6.1 | Dataset description

This section discusses about the dataset collected in real time known as network node characteristic features (NNCFs) dataset; after thorough empirical study, a dataset is prepared by using the Google Form,[19] and the form was filled by different people present at different geographical location. The Google Form was created to collect the node features namely RAM, CPU, IP address, MAC address, battery power utilization, and internal storage. The dataset was collected from August 23, 2019 to August 29, 2019. The overall dataset contains 114 node data with six unique features. There were lots of missing values hence, the random oversampling technique was used.[21] The screenshot of the collected dataset is shown in Figure 5, and the repository of the same is hosted for public access as free and open source software collection of records.[22]

6.1.1 | Random oversampling

This oversampling technique is widely used for balancing the imbalanced data in the dataset and to increase the number of samples in the dataset. One of the drawbacks of ROS is that it copies the data points which are already available, and therefore, it does not add any unique sample in the dataset.[23]

6.2 | Classification techniques

Classification technique is widely used to classify the data on the basis of its features into different classes. There are various machine learning classification techniques.[24] In this paper, three classification algorithms have been used as discussed in Section 6:

Multinomial logistic regression Multinomial logistic regression is also known as polytomous. When logistic regression analysis is performed beyond the dichotomous variable where there are more than two categories of dependent variable, then the resulting model is known as multinomial logistic regression.[25]

Support vector machine SVM is a classifier that classifies the data by a hyperplane, which is a line in two-dimensional space dividing the data into two classes.[26]

K-nearest neighbor KNN is used for the dataset which has complex features and is difficult to understand. It uses the K-nearest neighbors to classify the data in one of the class.[27]

The description of Figure 6 is as follows: The figure shows three networks, namely, MANET 1, MANET 2, and DTN, and inside MANET 1, the Bluetooth network is also present. There are eight nodes in MANET 1, three nodes in MANET 2, and two nodes in delay tolerant network. Each node in the network has a feature table which contains class, RAM, CPU, battery percentage, and internal storage of a particular node. Now, the node 1 of MANET 1 intends to communicate with node 2 of MANET 2 then the neighboring nodes of N1, namely, N2 and N3 are analyzed on the basis of their feature values present in their respective tables. The best hop node is selected, and the message is forwarded to that node and so on until the message is delivered to a destination node successfully.

The proposed work is simulated in RStudio where the network is created by using 10 nodes of NNCF dataset. On the basis of conditions mentioned in Table 2 of proposed work, the best hop node is selected, and the directed edge is formed between the two nodes. The network thus created when this simulation is performed is shown in Figure 7.[28]

The RStudio version 1.2.1335 has been used for the simulation of the proposed work.[29]

7 | MATHEMATICAL REPRESENTATION

7.1 | Relationship between node classification and node selection

To find out the best neighboring node or the best next hop, it involves two steps.

1. Identification of neighboring nodes.
2. Selection of best next hop node on the basis of selection conditions.

T(i) : a node $i \in$ [CLASS 1, CLASS 2, CLASS3] and $\mathbf{i_t}>1$

where i = neighboring node of a source node and

i_t = trust factor of node i

N = $\{N_1, N_2, N_3 ... N_n\}$ // set of all neighboring nodes of a source node

I = { empty set } // empty set of intermediate nodes between source and destination node.

$\exists$ N | T(N) then N is said to be best next hop node.

I = I + N

8 | METHODOLOGY ADOPTED

This section includes the methodology used in this research paper. Initially, the dataset was collected as shown in Figure 5 of Section 6. The actual dataset contains the missing values, and therefore, predicive mean matching (PMM) method have been used, an imputation method for missing values present in the dataset and for balancing the dataset the random oversampling technique have been used.

The algorithms are designed for the proposed methodology, and the description of each algorithm is given in Table 5. The complexity of the proposed algorithm is detailed in Equation (1) where N is the number of nodes in a dataset, M

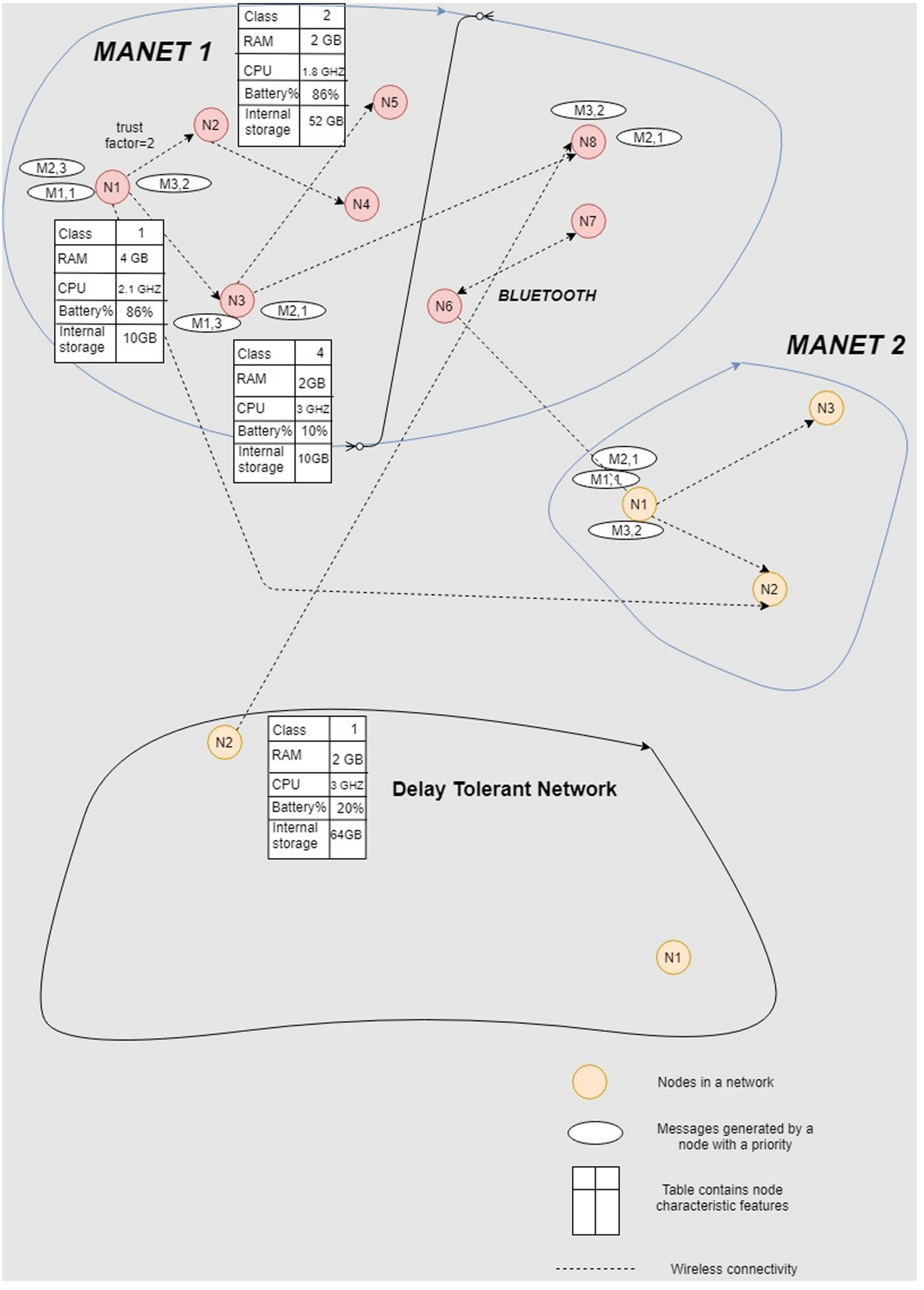

FIGURE 6 Network representation

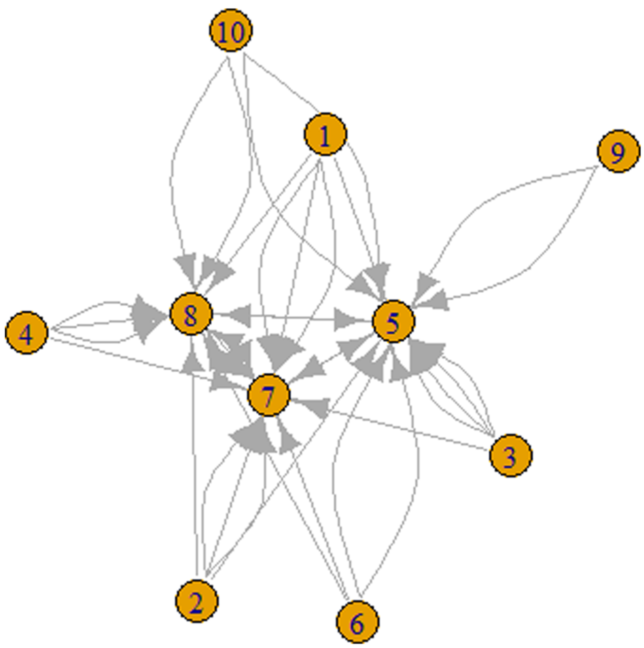

FIGURE 7 Graphical representation of a network nodes during simulation

TABLE 5 Description of algorithms

Algorithm	Algorithm name	Purpose/description	Time complexity
Algorithm 1	ORuML algorithm	Main algorithm of the proposed methodology	$(n_{sv}d)+(cd))$ $O(N^2+(nd+kn)+$
Algorithm 2	Node colocation algorithm	Algorithm to find out whether the source and destination nodes are colocated, that is, whether the destination node is a part of same network as source node	O(NM)
Algorithm 3	Network type identification algorithm	Algorithm to find out whether the source and destination nodes are part of homogeneous network or heterogeneous network	O(NM)
Algorithm 4	Trust factor algorithm	Algorithm to assign trust factor to each edge on the network	O(N)
Algorithm 5	Priority of a message algorithm	Algorithm to assign priority to each message generated by the source node	$O(N^2)$
Algorithm 6	Algorithm for KNN classification technique	Algorithm to apply KNN classification technique on the network node characteristic features dataset and to output the accuracy and area under ROC curve	O(nd+kn)
Algorithm 7	Algorithm for SVM classification technique	Algorithm to apply SVM classification technique on the network node characteristic features dataset and to output the accuracy and area under ROC curve	$O(n_{sv}d)$
Algorithm 8	Algorithm for MLR classification technique	Algorithm to apply MLR classification technique on the network node characteristic features dataset and to output the accuracy and area under ROC curve	O(cd)

is the range of random number generator, n is the number of training samples, d is the number of features, k is the hyperparameter, and c is the number of support vectors, and when simulation was performed, the value of hyperparameter variable k was 5, the value of support vector n_{sv} was 380, and the number of classes c was 3.

$$O(N^2 + (nd + kn) + (n^{sv}d) + (cd)) \tag{1}$$

Algorithm 1 ORuML Algorithm

Algorithm:

for(S_N,$D_N \in N$)
val=Algorithm 2 // To check (S_N, D_N) $\in$ SN or DN
if(val == 1) **then**
(Fn, Tn) $\in$ SN
else (Fn, Tn) $\in$ DN
val2 = Algorithm 3 // to check (SN, DN) $\in H_o$ or H_t
if(val2 = = 1) **then**
(F_n, T_n) $\in H_o$
else (Fn, Tn) $\in H_t$
for e $\in$ E
val3=Algorithm 4 //calculate trust factor
Algorithm 5 // assign priority to each message
Algorithm 6 // KNN algorithm applied on dataset
Algorithm 7 // SVM algorithm applied on dataset
Algorithm 8 // MLR algorithm applied on dataset

Algorithm 2 node colocation algorithm

Algorithm:

for i $\in$ [1,N]
for j $\in R_n$
if(C_n = = “CLASS 1” || C_n = = “CLASS 2”|| C_n = = “CLASS 3”) **then**
$G = G + G_e$
for $F_n \in S_n$ and $T_n \in S_n$
if N_{id} [F_n] = = N_{id}[T_n]) **then**
return 1
else
return 2

Algorithm 3 network type identification algorithm

Algorithm:

for i $\in$ [1,N]
for j $\in R_n$
if(C_n = = “BLUETOOTH” || C_n = = “MANET”|| C_n = = “DTN”) **then**
$G = G + G_e$
for $F_n \in S_n$ and $T_n \in S_n$
if N_t [F_n] = = N_t[T_n]) **then**
return 1
else return 0

Algorithm 4 Trust Factor algorithm

Algorithm:

for i $\in$ [1,N] // i $\in E_G$
for j $\in$ [i+1,N]
if(F_n [i] = = F_n [j] and T_n [i] = = T_n [j])
$T_f = T_f + 1$
return T_f

Algorithm 5 Algorithm for priority of a message

Algorithm:

for i $\in$ [1,N]
for j $\in R_n$ // j number of messages generated by the node i
Assign priority p to each j

Algorithm 6 Algorithm for KNN classification technique

Algorithm:

Input: $DS_{train} = n_1, n_2, n_3, \ldots ..n_k$ // Training dataset which includes training examples and their classes.
$DS_{test} = n_1, n_2, n_3, \ldots ..n_m$ // Test dataset which includes test examples and their classes
Output: KNN accuracy and AUC

Method:
STEP 1: Select a method and apply it on training dataset DS_{train}
STEP 2: Generate a model based on the KNN algorithm
STEP 3: Predict a model on test dataset DS_{test}
STEP 4: Output accuracy on predicted model
STEP 5: Output AUC on the predicted model

Algorithm 7 Algorithm for SVM classification technique

Algorithm:

Input: $DS_{train} = n_1, n_2, n_3, \ldots ..n_k$ // Training dataset which includes training examples and their classes.
$DS_{test} = n_1, n_2, n_3, \ldots ..n_m$ // Test dataset which includes test examples and their classes
Output: SVM accuracy and AUC

Method:
STEP 1: Select a method and apply it on training dataset DS_{train}
STEP 2: Generate a model based on the SVM algorithm
STEP 3: Predict a model on test dataset DS_{test}
STEP 4: Output accuracy on predicted model
STEP 5: Output AUC on the predicted model

Algorithm 8 Algorithm for MLR classification technique

Algorithm:

Input: $DS_{train} = n_1, n_2, n_3, \ldots ..n_k$ // Training dataset which includes training examples and their classes.
$DS_{test} = n_1, n_2, n_3, \ldots ..n_m$ // Test dataset which includes test examples and their classes
Output: MLR accuracy and AUC

Method:
STEP 1: Select a method and apply it on training dataset DS_{train}
STEP 2: Generate a model based on the MLR algorithm
STEP 3: Predict a model on test dataset DS_{test}
STEP 4: Output accuracy on predicted model
STEP 5: Output AUC on the predicted model

TABLE 6 Comparsion on the basis of accuracy and AUC

S. no	Techniques	Accuracy (%)	AUC (%)
1	SVM	78.95	72.59
2	MLR	94.74	93.42
3	KNN	92.98	79.20

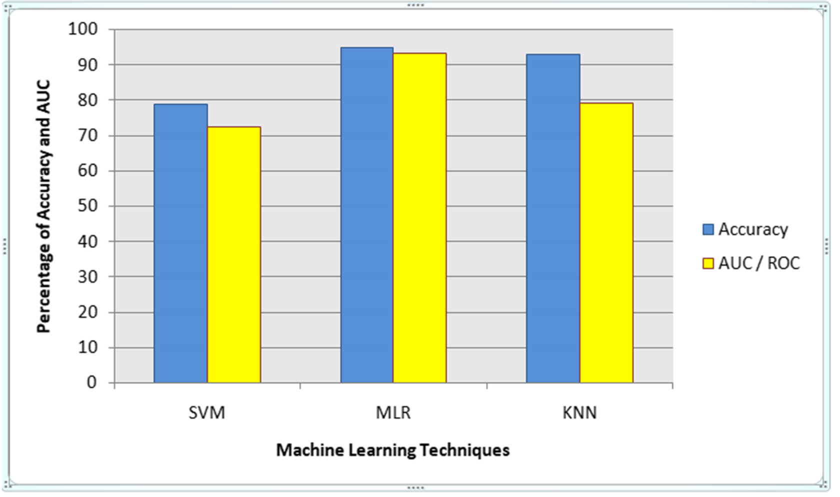

FIGURE 8 Performance graph for accuracy and AUC of KNN, MLR, and SVM

9 | RESULTS

This section includes the result of experiment performed on the network dataset of real time nodes in the network. The machine learning techniques applied were namely KNN, MLR and SVM on the dataset and the result was analyzed by using the performance metrics viz. Accuracy and AUC(Area Under ROC Curve) as shown in Table 6 . Schematically, the performance graph is depicted between machine learning techniques and the percentage of accuracy and AUC as shown in Figure 8. The observations of the performance over the resulted dataset obtained by applying the random over sampling technique are as follows:

1. Results shows that MLR outperformed the other techniques namely SVM and KNN in terms of accuracy and AUC, thus justifying our intuition.
2. The performance of SVM was not significant when compared to MLR and KNN.

10 | CONCLUSION AND FUTURE WORK

The current research work proposes a new ORuML algorithm for a wireless network. The machine learning technique, namely, KNN, MLR, and SVM has been applied on the NNCF dataset to predict the network type of the nodes and simulation highlight the fact that MLR outperforms in accuracy and AUC over KNN and SVM, and therefore, it predicts the best out of three algorithms used. Also, the best neighboring node selection has been simulated for efficient routing in a MANET and DTN, respectively.

As future extension of current research work, it is proposed that decision tree, random forest, and so forth machine learning algorithms can be applied on the given network node characteristic features (NNCFs) dataset. Apart from this, deep learning can be applied on the given dataset, and the same work can be extended to a cellular network. The machine learning can also be used for routing a message in an opportunistic network.

ACKNOWLEDGEMENTS

We would like to thank our institution Guru Gobind Singh Indraprastha University for such a great exposure to accomplish such tasks and providing a strong platform to develop our skills and capabilities.

ORCID

Sapna Chaudhary https://orcid.org/0000-0003-2623-5177

Rahul Johari https://orcid.org/0000-0002-7675-8550

REFERENCES

1. Bhatia N. Survey of nearest neighbor techniques. arXiv preprint arXiv:1007.0085; 2010.
2. Turing AM. book=Parsing the Turing Test, editor=Epstein, R, editor=Roberts, G, editor=Beber, G, address=Dordrecht, publisher=Springer Netherlands; 2009:23-65.
3. Russell B, Littman ML, Trappe W. Integrating machine learning in ad hoc routing: A wireless adaptive routing protocol. *Int J Commun Syst*. 2011;24(7):950-966.
4. Ghouti L. Mobility prediction in mobile ad hoc networks using neural learning machines. *Simul Modell Pract Theory*. 2016;66:104-121.
5. Jiang C, Zhang H, Ren Y, Han Z, Chen K-C, Hanzo L. Machine learning paradigms for next-generation wireless networks. *IEEE Wirel Commun*. 2016;24(2):98-105.
6. Sharma DK, Dhurandher SK, Woungang I, Srivastava RK, Mohananey A, Rodrigues JJ. A machine learning-based protocol for efficient routing in opportunistic networks. *IEEE Syst J*. 2016;12(3):2207-2213.
7. Ghaffari A. Real-time routing algorithm for mobile ad hoc networks using reinforcement learning and heuristic algorithms. *Wirel Netw*. 2017;23(3):703-714.
8. Wang M, Cui Y, Wang X, Xiao S, Jiang J. Machine learning for networking: Workflow, advances and opportunities. *IEEE Netw*. 2017;32(2):92-99.
9. Mao B, Fadlullah ZM, Tang F, et al. Routing or computing? the paradigm shift towards intelligent computer network packet transmission based on deep learning. *IEEE Trans Comput*. 2017;66(11):1946-1960.
10. Rath M, Pati B, Pattanayak BK, Panigrahi CR, Sarkar JL. Load balanced routing scheme for manets with power and delay optimisation. *Int J Commun Netw Distrib Syst*. 2017;19(4):394-405.
11. Roy A, Kar P, Misra S, Obaidat MS. D3: Distributed approach for the detection of dumb nodes in wireless sensor networks. *Int J Commun Syst*. 2017;30(1).
12. Li Z, He T. Webee: Physical-layer cross-technology communication via emulation. In: Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking ACM; 2017:2-14.
13. Musumeci F, Rottondi C, Nag A, et al. An overview on application of machine learning techniques in optical networks. *IEEE Commun Surv Tutor*. 2018;21(2):1383-1408.
14. Ayoubi S, Limam N, Salahuddin MA, et al. Machine learning for cognitive network management. *IEEE Commun Mag*. 2018;56(1):158-165.
15. Wang W, He S, Sun L, Jiang T, Zhang Q. Cross-technology communications for heterogeneous iot devices through artificial doppler shifts. *IEEE Trans Wirel Commun*. 2018;18(2):796-806.
16. Challita U, Ferdowsi A, Chen M, Saad W. Machine learning for wireless connectivity and security of cellular-connected uavs. *IEEE Wirel Commun*. 2019;26(1):28-35.
17. Anwar MZ, Kaleem Z, Jamalipour A. Machine learning inspired sound-based amateur drone detection for public safety applications. *IEEE Trans Veh Technol*. 2019;68(3):2526-2534.
18. Chaudhary S, Johari R, Bhatia R, Gupta K, Bhatnagar A. Craiot: Concept, review and application (s) of iot. In: 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (iot-siu) IEEE; 2019; Ghaziabad, India, India:1-4.
19. Chaudhary S. Google form. (accessed september 2019) http://bit.ly/2mgvHdb; 2019.
20. online. Nepalmap. (accessed January 10, 2020) https://wcvchurch.ca/hrv-google-earth-video-map/; 2020.
21. RStudio. documentation. (accessed October 2019) https://www.rdocumentation.org/packages/mlr/versions/2.15.0/topics/oversample; 2019.
22. GitHub. dataset. (accessed January 3, 2020) https://github.com/sapnachaudhary/Node-dataset/; 2020.
23. Branco P, Ribeiro RP, Torgo L. Ubl: An r package for utility-based learning. arXiv preprint arXiv:1604.08079; 2016.
24. Soofi AA, Awan A. Classification techniques in machine learning: Applications and issues. *J Basic Appl Sci*. 2017;13:459-465.
25. Menard S. *Logistic Regression: From Introductory to Advanced Concepts and Applications*: Sage; 2010.
26. Smola AJ, Schölkopf B. A tutorial on support vector regression. *Stat Comput*. 2004;14(3):199-222.
27. Lantz B. *Machine Learning With R*: Packt Publishing Ltd; 2013.
28. Luke DA. *A User's Guide to Network Analysis in R*: Springer; 2015.
29. RStudio. download. (accessed October 2019) https://www.rstudio.com/products/rstudio/download/; 2019.

How to cite this article: Chaudhary S, Johari R. ORuML: Optimized Routing in wireless networks using Machine Learning. *Int J Commun Syst*. 2020;e4394. https://doi.org/10.1002/dac.4394